The Human–Computer Interaction Handbook

Fundamentals,
Evolving Technologies,
and Emerging Applications

Third Edition

Edited by Julie A. Jacko, Ph.D.

CRC Press
Taylor & Francis Group

The Human–Computer Interaction Handbook

Fundamentals, Evolving Technologies, and Emerging Applications

Third Edition

Human Factors and Ergonomics

Series Editor

Gavriel Salvendy

Professor Emeritus	***Chair Professor & Head***
School of Industrial Engineering	*Dept. of Industrial Engineering*
Purdue University	*Tsinghua Univ., P.R. China*

PUBLISHED TITLES

Conceptual Foundations of Human Factors Measurement, *D. Meister*

Content Preparation Guidelines for the Web and Information Appliances: Cross-Cultural Comparisons
H. Liao, Y. Guo, A. Savoy, and G. Salvendy

Designing for Accessibility: A Business Guide to Countering Design Exclusion, *S. Keates*

Handbook of Cognitive Task Design, *E. Hollnagel*

The Handbook of Data Mining, *N. Ye*

Handbook of Digital Human Modeling: Research for Applied Ergonomics and Human Factors Engineering
V. G. Duffy

Handbook of Human Factors and Ergonomics in Health Care and Patient Safety, Second Edition,
P. Carayon

Handbook of Human Factors in Web Design, Second Edition, *R. Proctor and K. Vu*

Handbook of Occupational Safety and Health, *D. Koradecka*

Handbook of Standards and Guidelines in Ergonomics and Human Factors, *W. Karwowski*

Handbook of Virtual Environments: Design, Implementation, and Applications, *K. Stanney*

Handbook of Warnings, *M. Wogalter*

Human–Computer Interaction: Designing for Diverse Users and Domains, *A. Sears and J. A. Jacko*

Human–Computer Interaction: Design Issues, Solutions, and Applications, *A. Sears and J. A. Jacko*

Human–Computer Interaction: Development Process, *A. Sears and J. A. Jacko*

The Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies,
and Emerging Applications, Second Edition, *A. Sears and J. A. Jacko*

Human Factors in System Design, Development, and Testing, *D. Meister and T. Enderwick*

Introduction to Human Factors and Ergonomics for Engineers, *M. R. Lehto and J. R. Buck*

Macroergonomics: Theory, Methods and Applications, *H. Hendrick and B. Kleiner*

Practical Speech User Interface Design, *James R. Lewis*

Smart Clothing: Technology and Applications, *Gilsoo Cho*

Theories and Practice in Interaction Design, *S. Bagnara and G. Crampton-Smith*

The Universal Access Handbook, *C. Stephanidis*

Usability and Internationalization of Information Technology, *N. Aykin*

User Interfaces for All: Concepts, Methods, and Tools, *C. Stephanidis*

FORTHCOMING TITLES

Computer-Aided Anthropometry for Research and Design, *K. M. Robinette*

Cross-Cultural Design for IT Products and Services, *P. Rau, T. Plocher and Y. Choong*

Foundations of Human–Computer and Human–Machine Systems, *G. Johannsen*

The Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies,
and Emerging Applications, Third Edition, *J. A. Jacko*

Handbook of Virtual Environments: Design, Implementation, and Applications, Second Edition
K. S. Hale and K M. Stanney

Introduction to Human Factors and Ergonomics for Engineers, Second Edition, *M. R. Lehto*

The Science of Footwear, *R. S. Goonetilleke*

Skill Training in Multimodal Virtual Environments, *M. Bergamsco, B. Bardy, and D. Gopher*

The Human–Computer Interaction Handbook

Fundamentals, Evolving Technologies, and Emerging Applications

Third Edition

Edited by Julie A. Jacko, Ph.D.

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

Cover art by Torre Schmitt (Torre Schmitt Design, Minneapolis, MN, torreschmitt@gmail.com).

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20120227

International Standard Book Number-13: 978-1-4398-2944-8 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

This handbook is dedicated to those who have lent a hand and lit the way.

And I said to the man who stood at the gate of the year:

"Give me a light, that I may tread safely into the unknown!"

And he replied:

"Go out into the darkness and put your hand into the Hand of God.

That shall be to you better than light and safer than a known way."

King George VI in his New Year's message to his embattled people
at the beginning of the Second World War

Contents

PART I Humans in HCI

PART II Computers in HCI

PART III Designing Human–Computer Interactions

PART IV Application-/Domain-Specific Design

PART V Designing for Diversity

PART VI The Development Process
SECTION A Requirements Specification

SECTION B Design and Development

SECTION C Testing, Evaluation, and Technology Transfer

PART VII Emerging Phenomena in HCI

Series Foreword

The third edition of this classic handbook is published at an opportune time when interactive technologies are a dominating presence in work, leisure, and social settings and when ambient intelligence is gaining accelerated momentum. The field of human–computer interaction (HCI) has matured to such an extent that even the words comprising the term have taken on new, expanded, and reinterpreted meanings. That is, the field has advanced significantly from its origins. Researchers in HCI are called upon now more than ever to develop new knowledge, which often resides at the intersection of multiple disciplines and spans various and innovative platforms of applications. Information technology is more ubiquitous today than ever, successfully interacting with the technologies that ensure it is more enjoyable and more productively accessible and usable by all segments of society across all five continents.

This handbook is the premier resource for the theoretical and operational foundations of HCI, providing readers access to the latest scientific breakthroughs coupled with the state of the art in the field. The book provides detailed descriptions of approaches and methodologies that are frequently illustrated with case studies and examples on how to conceptualize, design, and evaluate interactive systems with human beings at the center of the endeavor. As such, this handbook will be invaluable to researchers, practitioners, educators, and students working in, or at the intersection of, computer science, information technology, information science, informatics, engineering, psychology, design, and human factors and ergonomics.

This book is part of the Human Factors and Ergonomics series, published by the Taylor & Francis Group. The 145 authors of this handbook include 92 from academia, 49 from industry, and 4 from government agencies. These individuals are among the very best and most respected in their fields across the globe. The more than 80 tables, 400 figures, and nearly 7000 references in this book provide the single most comprehensive depiction of this field that exists in a single volume.

The handbook authors come from 14 countries: Australia, Canada, Cyprus, Denmark, Finland, France, Germany, Greece, Ireland, Japan, South Africa, the Netherlands, United Kingdom, and the United States.

Gavriel Salvendy, Series Editor
Purdue University/Tsinghua University, China

Foreword

The Expanding Impact of Human–Computer Interaction

The remarkable growth of human–computer interaction (HCI) over the past 30 years has transformed this nascent interdisciplinary field into an intellectually rich and high impact worldwide phenomenon. We have grown from a small rebellious group of researchers who struggled to gain recognition as we broke disciplinary boundaries to a broad influential community with potent impact on the daily lives of every human. There are dozens of relevant journals, plus conferences and workshops worldwide.

The aspirations of early HCI researchers and practitioners were to make better menus, design graphical user interfaces based on direct manipulation, improve input devices, design effective control panels, and present information in comprehensible formats. HCI software developers contributed innovative tools that enabled programmers and nonprogrammers to create interfaces for widely varying applications and diverse users. HCI professionals developed design principles, guidelines, and sometimes standards dealing with consistency, informative feedback, error prevention, shortcuts for experts, and user control. Success was measured by individual performance metrics such as learning time, speed, error rates, and retention for specific tasks, whereas user satisfaction was assessed by detailed questionnaires filled with numbered scales.

In the early days, HCI researchers and professionals fought to gain recognition and often still have to justify HCI's value with academic colleagues or corporate managers. However, the larger world embraced our contributions and now has high expectations of what we can deliver. Few fields can claim such rapid expansion and broad impact as those who design the desktop, web, mobile, and cellphone interfaces that have spread around the world into the hands of at least 5 billion users. HCI designs now influence commercial success, reform education, change family life, affect the political stability of nations, are embedded in military systems and play a significant role in shaping a peaceful or conflict-ridden world.

The Handbook of Human–Computer Interaction: Third Edition details the progress of this extraordinary discipline, inviting newcomers to learn about it and helping experienced professionals to understand the rapid and continuing changes. The carefully written chapters and extensive references will be useful to readers who want to scan the territory or dig deep into specific topics. This handbook's prominent authors thoughtfully survey the key topics, enabling students, researchers, and professionals to appreciate HCI's impact.

As HCI progresses, there is a greater acceptance in the academic environment, where HCI is now part of most computer science, iSchool, business, engineering, and other departments and has advocates in medicine, social sciences, journalism, humanities, etc. Although the term *human–computer interaction* has achieved widespread recognition, many insiders feel that it is no longer an accurate description. They complain that it suggests one human interacting with one computer to complete narrow tasks. Instead, these critics believe that the discipline should reflect user-oriented technologies that are ubiquitous, pervasive, social, embedded, tangible, invisible, multimodal, immersive, augmented, or ambient. Some want to break free from the focus on computer use and emphasize user experiences, interaction design, emotional impact, aesthetics, social engagement, empathic interactions, trust building, and human responsibility.

New terms have been proposed such as *human-centered computing, social computing, human–information interaction, human–social interaction, human-centered informatics,* or just *human interaction*. Novel, but already thriving applications areas include computational biology, computational social science, e-commerce (and m-commerce), digital humanities, information visualization, open government, sustainability, biodiversity, and citizen science. Although these broader visions are important, many researchers are still working on innovative display designs, input devices, multimedia output, programming toolkits, and predictive models of user performance.

New names and applications are a good sign of success, but finding the balance between sticking with an established term and welcoming innovative directions is difficult. Maybe an old aphorism helps: "make new friends and keep the old, one is silver and the other gold." Can we retain the brand name recognition of HCI but embrace new directions by discussing *micro-HCI* and *macro-HCI*?

Micro-HCI researchers and developers would design and build innovative interfaces and deliver validated guidelines for use across the range of desktop, web, mobile, and ubiquitous devices. The challenges for micro-HCI are to deal with rapidly changing technologies, while accommodating the wide range of users: novice/expert, young/old, literate/illiterate, abled/disabled, and their cultural plus linguistic diversity. These distinctions are tied to skills, but there are further diversities in gender, personality, ethnicity, skills, and motivation that are now necessary to address in interface designs. Micro-HCI researchers can take comfort in dealing with well-stated requirements, clear benchmark tasks, and effective predictive models.

Macro-HCI researchers and developers would explore new design territories such as affective experience, aesthetics, motivation, social participation, trust, empathy, responsibility, and privacy. The challenges for macro-HCI are to deal with new opportunities across the range of human experience: commerce, law, health/wellness, education, creative arts, community relationships, politics, policy negotiation, conflict resolution, international development, and peace studies. Macro-HCI researchers have to face the challenge of more open tasks, unanticipated user goals, and even conflicts among users in large communities.

Although micro-HCI and macro-HCI have healthy overlaps, as do micro-economics and macro-economics, they attract different types of researchers, practitioners, and activists, thereby further broadening the scope and impact. As commercial, social, legal, and ethical considerations play an increasing role, educational curricula and professional practices need to be updated regularly and midcareer continuing education for HCI professionals will keep them current.

An important goal will be to develop new metrics and evaluation methods for micro-HCI and macro-HCI. Moore's Law has been useful in charting the growth of computing, enabling everyone to admire and benefit from the increase in gigahertz, terabytes, and petaflops. These are still useful, but we need newer metrics to understand the impact of HCI designs that have enabled the spread of billions of mobile devices and the emergence of YouTube, Facebook, twitter, Wikipedia, and so on. Understanding this transformation would be facilitated by measures of giga-hellos, teracontribs, and peta-thankyous and by newer metrics of trust, empathy, responsibility, privacy, and so on.

Traditional evaluation approaches of controlled experiments and usability testing are being continuously refined to fit the needs of micro-HCI, whereas the newer methods of qualitative, ethnographic, and case study methods are being explored to match the needs of macro-HCI. Both groups will benefit from the remarkable increased opportunities to log usage on a massive scale through the increasingly connected communications, data, and sensor networks. Traditional surveys of a small sample of users who offer biased perceptions or reports of attitudes are giving way to actual measurement of usage that reveals the learnability, efficacy, utility, and satisfaction of users. Even more exciting is the potential to capture the manifestations of trust, empathy, responsibility, privacy, security, and motivation. Researchers are also beginning to measure brand loyalty, parental engagement, political leaning, potential for violence, community commitment, and much more. The dangers of inappropriate intrusion, misguided applications, scamming/spamming, deception, and bullying are now part of macro-HCI. Even greater concerns come from criminals, terrorists, and oppressive governments who can use these technologies in ways that threaten individuals, intimidate communities, or destroy the environment.

The power of widely used social technologies that stem from HCI's success means that we will face ethical challenges similar to what the nuclear physicists dealt with during the 1940s and beyond. We cannot and should not avoid these responsibilities. Rather, we should embrace them and show leadership in shaping technology to produce positive outcomes. This is never easy, but every worthy project that improves the health, environment, or education of children or builds capacity for constructive communities should be recognized, disseminated, scaled up, and continuously improved. Even more ambitious should be our efforts to promote open government, independent oversight, deliberative systems, and citizen participation. The research agenda for HCI should include the UN Millennium Development Goals such as eradicating extreme hunger and poverty, ensuring universal childhood education, promoting maternal health, and ensuring environmental sustainability. If HCI professionals also courageously address conflict resolution, international development, and peace studies, we can inspire others and help build a better world.

We should be proud of what HCI has accomplished, but there is much work to be done. Let's get on with it!

ACKNOWLEDGMENTS

Thanks to Ron Baecker, Jack Carroll, Susan Dray, Gerhard Fischer, Rob Jacob, Clare-Marie Karat, Clayton Lewis, Brad Myers, and Jenny Preece for comments on earlier drafts.

Ben Shneiderman
University of Maryland

Preface

This third edition of the HCI handbook represents the single largest, most complete compilation of HCI theories, principles, advances, case studies, and more that exist within a single volume. The construction of the handbook has been a massive community effort of which it was a tremendous privilege for this author and editor to be a part. The 145 authors of the 62 chapters within this book are people who have not only dedicated themselves to laying the foundation for this field but also dared to address the grand challenges that have been posed along the way, thus advancing the field of HCI by leaps and bounds. The HCI community from which these authors hail is remarkably diverse and collaborative. You will see the artifacts of this ethos throughout the book.

The handbook opens with an insightful and thought-provoking introduction written by Jonathan Grudin, which sets the tone for the entire book. Within the introduction you will find a unique and compelling depiction of the evolution of HCI. The handbook closes with a look at the evolving nature of HCI to change the world. The closing chapter is written by the largest collection of authors in the book, led by Susan Dray. The global focus of this chapter is personified by the authors' origins, which literally span the globe. The chapters in between are organized very much like those in the second edition; however, the content of the chapters has been dramatically updated to reflect the state of the art and current state of the science in HCI. There have been numerous notable additions to the third edition, which reflect the ever-growing nature of this field, including, for example, chapters on social networks and social media, grounded theory, choices and decisions of users, and the naturalistic approach to evaluation.

I offer my heartfelt thanks to Ben Shneiderman, who kindly agreed to contribute his revolutionary perspective in the Foreword to the third edition. He not only chronicles the impact of HCI but also presents a challenge to each and every one of us to embrace the responsibility of shaping technology to produce positive outcomes. With this challenge he is asking us to be the best citizen scholars we can be. This is classic Ben Shneiderman and just one of the many reasons why I respect and admire him. This handbook would simply not have been possible without the guiding influence of my longtime mentor and good friend, Gavriel Salvendy. Gavriel sets the standard for successfully coalescing people and communities around shared goals and mutual aspirations. He has been an unwavering source of inspiration, support, advice, opportunity, and kindness for me. This book is part of a larger book series of which Gavriel is the series editor. His Series Foreword to the third edition enables us to see this book in the context of the larger whole. Both these luminaries, Ben and Gavriel, have transformed the field of HCI in their own signature ways, and I salute both of them.

A very special individual worked hand in hand with me in constructing the third edition. Molly McClellan, PhD, is a research associate with SimPORTAL at the University of Minnesota, performing postdoctoral research in the area of perioperative simulation. Completing a book of this scale and scope requires incredible persistence and perseverance. Molly demonstrates both these attributes and so much more. She is a creative problem solver with an uncanny ability to organize vast quantities of information from disparate and geographically distributed sources. She is smart, generous, and exceedingly committed to excellence. I have admired her as a scholar and as a human being. It is a privilege to serve as her major professor and mentor.

Last but not the least, I wish to recognize the support offered me by my husband François and our son Nico. They are both, quite simply, my *raison de vivre*.

Julie A. Jacko
University of Minnesota

Editor

Julie A. Jacko, PhD, is a professor of Public Health at the University of Minnesota and a faculty fellow in the Academic Health Center's Institute for Health Informatics. She is the principal investigator and director of the University Partnership for Health Informatics (UP-HI). This $5.1-million grant from the Office of the National Coordinator for Health Information Technology is one of the nine awarded nationally and represents the first public–private partnership funded in the upper Midwest United States to infuse our nation's workforce with individuals who have been trained to perform one of six mission-critical health information technology roles.

Dr. Jacko has expertise in the design, implementation, and evaluation of interactive computing systems in complex domains such as population health and health care delivery, with the purpose of enhancing human performance and satisfaction. This is accomplished through research that is focused on the cognitive processes underlying the interaction of people with complex systems with the ultimate goal of combining robust empirical results with the development of engineering models of human performance that can aid in the design of real-world systems. Dr. Jacko has an exemplary research track record spanning nearly 20 years during which over 160 scientific publications have been generated in these research areas. She has generated nearly $25 million by way of research funding in the last 10 years and is one of the only 20 recipients of a National Science Foundation Presidential Early Career Award for Scientists and Engineers, the highest honor awarded to young investigators by the U.S. government. Dr. Jacko served as co-author on the #1-rated published article for 2005 in the *International Journal of Medical Informatics*. She has an extensive track record of professional leadership excellence, including the following awards and honorable positions.

- Received commendation from the Office of the National Coordinator for Health Information Technology (ONC) for her innovation and leadership in the University Partnership for Health Informatics (2011).
- Ranked one of the Top Ten Influential Informatics Professors by HealthTechTopia on September 14, 2010. (http://mastersinhealthinformatics.com/2010/top-10-most-influential-informatics-professors/).
- Published an invited comment paper in 2011, Issue 470, in *Nature* titled, "Narrow the Gap in Health Literacy."
- Appointed by the State of Minnesota Commissioner of Health to serve on the Minnesota e-health Advisory Committee, representing academics and clinical research for the State of Minnesota.
- Elected to the Office of the President, the Association for Computing Machinery's Special Interest Group on Computer–Human Interaction (ACM SIGCHI) (2006–09).
- Elected to the Association for Computing Machinery Special Interest Group Governing Board Executive Committee, elected Member at Large (2007–2009).
- Elected to the Office of the Vice President for Membership and Communications—the Association for Computing Machinery's Special Interest Group on Computer–Human Interaction (ACM SIGCHI) (2003–06).
- Editor-in-chief of the *International Journal of Human–Computer Interaction*, published by Taylor & Francis.
- Co-editor of the 1st and 2nd Editions of the *Human–Computer Interaction Handbook*, the premier compendium of research and practice in the field of human–computer interaction.

In addition, during the last 15 years, Dr. Jacko has chaired or co-chaired numerous technical conferences and technical conference programs in the fields of human factors and human–computer interaction. She received her BS, MS, and PhD in industrial and systems engineering from Purdue University in West Lafayette, Indiana, where she held the NEC Graduate Fellowship.

Contributors

Tamara Adlin
Adlin, Inc.
Seattle, Washington

Norman Alm
School of Computing
University of Dundee
Dundee, Scotland

Chee Siang Ang
School of Engineering and Digital Arts
University of Kent
Canterbury, United Kingdom

Helen Ashman
School of Computer and Information
Science
University of South Australia
Adelaide, Australia

Alisa Bandlow
Sandia National Laboratories
Albuquerque, New Mexico

Roger Beatty
Operational Control Services
Dallas/Fort Worth, Texas

Michel Beaudouin-Lafon
Department of Computer Science
Université Paris-Sud
Paris, France

Jeanette Blomberg
IBM Almaden Research Center
San Jose, California

Tim Brailsford
School of Computer Science and
Information Technology
University of Nottingham
Nottingham, United Kingdom

Stephen Brewster
Department of Computing Science
University of Glasgow
Glasgow, Scotland

Carolyn Brodie
Brodie Consulting Group
Indianola, Iowa

Amy Bruckman
College of Computing
Georgia Institute of Technology
Atlanta, Georgia

Mark Burrell
Endeca Technologies
Cambridge, Massachusetts

Gaëlle Calvary
Grenoble Informatics Laboratory
Institut Polytechnique de Grenoble
Grenoble, France

Pascale Carayon
Department of Industrial and Systems
Engineering
Center for Quality and Productivity
Improvement (CQPI)
University of Wisconsin
Madison, Wisconsin

Stuart Card
Xerox PARC
Palo Alto, California

Alex Carmichael
School of Computing
University of Dundee
Dundee, Scotland

John M. Carroll
College of Information Sciences and
Technology
Pennsylvania State University
University Park, Pennsylvania

Sanjay Chandrasekharan
School of Interactive Computing
Georgia Institute of Technology
Atlanta, Georgia

Romeo Chua
School of Human Kinetics
University of British Columbia
Vancouver, British Columbia, Canada

Gilbert Cockton
School of Design
Northumbria University
New Castle, United Kingdom

Joseph V. Cohn
Office of Naval Research
Arlington, Virginia

Catherine Courage
Citrix Systems
Fort Lauderdale, Florida

Joëlle Coutaz
Grenoble Informatics Laboratory
Université Joseph Fourier
Grenoble, France

Sara J. Czaja
Department of Psychiatry and
Behavioral Sciences
Center on Aging
University of Miami Miller School of
Medicine
Miami, Florida

Declan Dagger
Department of Computer Science
Trinity College
Dublin, Ireland

Andrew M. Dearden
Communication and Computing
Research Centre
Sheffield Hallam University
South Yorkshire, United Kingdom

Melissa Densmore
School of Information
University of California
Berkeley, California

Jill Dimond
College of Computing
Georgia Institute of Technology
Atlanta, Georgia

Alan Dix
Talis
Birmingham, United Kingdom

Michael C. Dorneich
Human-Centered Systems
Honeywell Laboratories, Golden Valley
Minneapolis, Minnesota

Susan M. Dray
Dray and Associates, Inc.
Minneapolis, Minnesota

Allison Druin
Human–Computer Interaction Lab
College of Information Studies
University of Maryland
College Park, Maryland

Joseph S. Dumas
Dumas Consulting
Yarmouth Port, Massachusetts

Paula J. Edwards
HIMFormatics, LLC
Atlanta, Georgia

Vanessa Evers
Department of Electrical Engineering and Computer Science
University of Twente
Enschede, the Netherlands

Todd J. Follansbee
Web Marketing Resource
West Tisbury, Massachusetts

Andrea Forte
College of Information Science and Technology
Drexel University
Philadelphia, Pennsylvania

Jean E. Fox
Bureau of Labor Statistics
Washington, D.C.

Erik Frøkjær
Department of Computer Science
University of Copenhagen
Copenhagen, Denmark

Thomas Fuller
Microsoft Studios
Redmond, Washington

Krzysztof Z. Gajos
Harvard School of Engineering and Applied Sciences
Harvard University
Cambridge, Massachusetts

Norman D. Geddes
Applied Systems Intelligence, Inc.
Alpharetta, Georgia

Emilie W. Gould
Department of Communication
State University of New York at Albany
Albany, New York

James Goulding
School of Computer Science and Information Technology
University of Nottingham
Nottingham, United Kingdom

Paul A. Green
Driver Interface Group
Transportation Research Institute
University of Michigan
Ann Arbor, Michigan

Peter Gregor
School of Computing
University of Dundee
Dundee, Scotland

William M. Gribbons
Department of Human Factors
Bentley University
Waltham, Massachusetts

Jonathan Grudin
Microsoft Research
Redmond, Washington

Daniel V. Gunn
Microsoft Studios
Redmond, Washington

Kelly S. Hale
Design Interactive, Inc.
Oviedo, Florida

Gabriella M. Hancock
Department of Applied Physiology and Kinesiology
University of Florida
Gainesville, Florida

Peter A. Hancock
Department of Psychology
and
Institute for Simulation and Training
University of Central Florida
Orlando, Florida

Vicki L. Hanson
School of Computing
University of Dundee
Dundee, Scotland

Caroline C. Hayes
Department of Mechanical Engineering
University of Minnesota
Minneapolis/St. Paul, Minnesota

Ken Hinckley
Microsoft Research
Redmond, Washington

Jesse Hoey
School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada

Eve Hoggan
Department of Computer Science
University of Helsinki
Helsinki, Finland

Karen Holtzblatt
InContext Design
Concord, Massachusetts

Kasper Hornbæk
Department of Computer Science
University of Copenhagen
Copenhagen, Denmark

Hiroshi Ishii
MIT Media Lab
Massachusetts Institute of Technology
Cambridge, Massachusetts

Hiroo Iwata
Graduate School of Systems and Information Engineering
University of Tsukuba
Tsukuba, Japan

Julie A. Jacko
School of Public Health Division of Environmental and Health Sciences
University of Minnesota
Minneapolis, Minnesota

Jhilmil Jain
Google
Mountain View, California

Anthony Jameson
DFKI, German Research Center for Artificial Intelligence
Saarbrücken, Germany

Layne M. Johnson
Health Sciences Library and Institute for Health Informatics
University of Minnesota
Minneapolis, Minnesota

Matthew Kam
Human–Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania

Clare-Marie Karat
Karat Consulting Group
Aptos, California

John Karat
Karat Consulting Group
Aptos, California

Kevin Keeker
Zynga, Inc.
San Francisco, California

David Kieras
Electrical Engineering and Computer Science Department
University of Michigan
Ann Arbor, Michigan

Sandra Kogan
IBM Corporation
Cambridge, Massachusetts

Andrew Laghos
Department of Multimedia and Graphic Arts
Cyprus University of Technology
Lemesos, Cyprus

Jennifer Lai
IBM T. J. Watson Research Center
Yorktown Heights, New York

Steven J. Landry
School of Industrial Engineering
Purdue University
West Lafayette, Indiana

Adam Larson
United States Air Force

Nicole Lazzaro
XEODesign, Inc.
Oakland, California

Chin Chin Lee
Center on Aging
University of Miami Miller School of Medicine
Miami, Florida

V. Kathlene Leonard
Cloverture, LLC
Smyrna, Georgia

Ann Light
Communication and Computing Research Centre
Sheffield Hallam University
Sheffield, United Kingdom

Wendy E. Mackay
INRIA Saclay
Orsay, France

Aaron Marcus
Aaron Marcus and Associates, Inc.
Berkeley, California

Gary Marsden
Department of Computer Science
University of Cape Town
Cape Town, South Africa

Deborah J. Mayhew
Deborah J. Mayhew & Associates
West Tisbury, Massachusetts

Molly A. McClellan
SimPORTAL Medical School
University of Minnesota
Minneapolis, Minnesota

Alexander Mertens
Institute of Industrial Engineering and Ergonomics
RWTH Aachen University
Aachen, Germany

Michael J. Muller
IBM Research
Cambridge, Massachusetts

Alan F. Newell
School of Computing
University of Dundee
Dundee, Scotland

Heather Neyedli
Faculty of Kinesiology and Physical Education
University of Toronto
Toronto, Ontario, Canada

Gary M. Olson
School of Information and Computer Sciences
University of California
Irvine, California

Judith S. Olson
School of Information and Computer Sciences
University of California
Irvine, California

Declan O'Sullivan
Department of Computer Science
Trinity College
Dublin, Ireland

Sharon Oviatt
Incaa Designs
Seattle, Washington

A. Ant Ozok
Department of Information Systems
University of Maryland, Baltimore County
Baltimore, Maryland

Randy J. Pagulayan
Microsoft Studios
Redmond, Washington

Stephen J. Payne
University of Manchester
Manchester, United Kingdom

Robert W. Proctor
Department of Psychological Sciences
Purdue University
West Lafayette, Indiana

John Pruitt
Microsoft Corporation
Redmond, Washington

Graham Pullin
School of Computing
University of Dundee
Dundee, Scotland

Divya Ramachandran
Computer Science Division
and
Berkeley Institute of Design
University of California
Berkeley, California

Matthew Ray
Faculty of Kinesiology and Physical Education
University of Toronto
Toronto, Ontario, Canada

Margaret Re
Visual Arts Department
University of Maryland, Baltimore County
Baltimore, Maryland

Janice (Ginny) Redish
Redish & Associates, Inc.
Bethesda, Maryland

John T. Richards
IBM T. J. Watson Research Center
Yorktown Heights, New York

Ramon L. Romero
Interactive Entertainment Business
Microsoft Corporation
Redmond, Washington

Mary Beth Rosson
College of Information Sciences and Technology
Pennsylvania State University
University Park, Pennsylvania

François Sainfort
School of Public Health
University of Minnesota
Minneapolis, Minnesota

Nithya Sambasivan
Department of Informatics
University of California
Irvine, California

Anthony Savidis
ICS-FORTH and Department of Computer Science
University of Crete
Crete, Greece

Christopher M. Schlick
Institute of Industrial Engineering and Ergonomics
RWTH Aachen University
Aachen, Germany

and

Fraunhofer-Institute for Communication, Information Processing, and Ergonomics
Wachtberg, Germany

Jan-Felix Schmakeit
School of Computer and Information Science
University of South Australia
Adelaide, Australia

Dylan D. Schmorrow
U.S. Navy
Arlington, Virginia

Kevin M. Schofield
Microsoft Research
Redmond, Washington

Ingrid U. Scott
Pennsylvania State College of Medicine
University Park, Pennsylvania

David Siegel
Dray and Associates, Inc.
Minneapolis, Minnesota

Daniel Siewiorek
Human–Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania

Asim Smailagic
Institute for Complex Engineered Systems
Carnegie Mellon University
Pittsburgh, Pennsylvania

Michael J. Smith
Department of Industrial and Systems Engineering
Center for Quality and Productivity Improvement
University of Wisconsin
Madison, Wisconsin

Philip J. Smith
Cognitive Systems Engineering Laboratory
The Ohio State University
Columbus, Ohio

Thomas Smyth
School of Interactive Computing
Georgia Institute of Technology
Atlanta, Georgia

Kay M. Stanney
Design Interactive, Inc.
Oviedo, Florida

Thad Starner
College of Computing
Georgia Institute of Technology
Atlanta, Georgia

Constantine Stephanidis
ICS-FORTH and Department of Computer Science
University of Crete
Crete, Greece

Osamuyimen Stewart
IBM T. J. Watson Research Center
Yorktown Heights, New York

Marco Susani
Koz Susani Design
Chicago, Illinois

Alistair Sutcliffe
University of Manchester
Manchester, United Kingdom

James L. Szalma
Department of Psychology
University of Central Florida
Orlando, Florida

John C. Thomas
IBM T. J. Watson Research Center
Yorktown Heights, New York

Brygg Ullmer
Department of Computer Science
and
Center for Computation and Technology
Louisiana State University
Baton Rouge, Louisiana

Darelle van Greunen
Department of Information Technology
School of Information Technology
Nelson Mandela University
Port Elizabeth, South Africa

Kim-Phuong L. Vu
Department of Psychology
California State University
Long Beach, California

Vincent Wade
School of Computer Science and Statistics
Trinity College
Dublin, Ireland

Annalu Waller
School of Computing
University of Dundee
Dundee, Scotland

Suzanne Watzman
Watzman Information Design
Somerville, Massachusetts

Daniel J. Weeks
Department of Psychology
University of Lethbridge
Lethbridge, Alberta, Canada

Timothy N. Welsh
Faculty of Kinesiology and Physical Education
University of Toronto
Toronto, Ontario, Canada

Daniel Wigdor
Department of Computer Science
University of Toronto
Toronto, Ontario, Canada

Andrew D. Wilson
Microsoft Research
Redmond, Washington

Carsten Winkelholz
Fraunhofer-Institute for Communication, Information Processing and Ergonomics
Wachtberg, Germany

Niall Winters
London Knowledge Lab
Institute of Education
University of London
London, United Kingdom

Dennis Wixon
Startup Business Group
Microsoft Corporation
Redmond, Washington

Alan Woolrych
Department of Computing, Engineering, and Technology
University of Sunderland
United Kingdom

Nicole Yankelovich
Open Wonderland Foundation
Weston, Massachusetts

Panayiotis Zaphiris
Department of Multimedia and Graphic Arts
Cyprus University of Technology
Limassol, Cyprus

Martina Ziefle
Communication Science
Human–Computer Interaction Center
RWTH Aachen University
Aachen, Germany

Introduction

A Moving Target: The Evolution of Human–Computer Interaction

Jonathan Grudin

PREAMBLE: HISTORY IN A TIME OF RAPID OBSOLESCENCE

> "What is a typewriter?" my six-year-old daughter asked.
> I hesitated. "Well, it's like a computer," I began.

Why Study the History of Human–Computer Interaction?

A paper widely read 20 years ago concluded with the advice to design a word processor by analogy to something familiar to everyone: a typewriter. Even then, one of my Danish students questioned this reading assignment noting that "the typewriter is a species on its last legs." For most of the computing era, interaction involved 80-column punch cards, paper tape, line editors, 1920-character displays, 1-megabyte diskettes, and other extinct species. Are the interaction issues of those times relevant today? No.

Of course, aspects of the human side of human–computer interaction (HCI) change very slowly if at all. Much of what was learned about our perceptual, cognitive, social, and emotional processes when we interacted with older technologies applies to our interaction with emerging technologies as well. Aspects of how we organize and retrieve information persist, even as the specific technologies that we use change. The handbook chapters lay out relevant knowledge of human psychology; how and when that was acquired may not be critical and is not the focus here.

Nevertheless, there is a case for understanding the field's history, and the rapid pace of change may strengthen it:

- Several disciplines are engaged in HCI research and application, but few people are exposed to more than one. By seeing how each has evolved, we can identify possible benefits of expanding our focus and obstacles to doing so.
- Celebrating the accomplishments of past visionaries and innovators is part of building a community and inspiring future contributors, even when some past achievements are difficult to appreciate today.
- Some visions and prototypes were quickly converted to widespread application, whereas others took decades and some remain unrealized to this day. By understanding the reasons for different outcomes, we can assess today's visions more realistically.
- Crystal balls are notoriously unreliable, but anyone planning or managing a career in a rapidly changing field must consider the future. Our best chance to anticipate change is to find trajectories that extend from the past to the present. One thing is certain: The future will not resemble the present.

This account does not emphasize engineering "firsts." It focuses on technologies and practices as they became widely used, reflected in the spread of systems and applications. This was often paralleled by the formation of new research fields and changes in existing disciplines, which were marked by the creation and evolution of professional associations and publications. More a social history than a conceptual history, this survey points to trends and trajectories you might download into your crystal balls.

A historical account is a perspective. It emphasizes some things while de-emphasizing or omitting others. A history can be wrong in details, but is never right in any final sense. Your questions and your interests will determine how useful a perspective is to you. This introduction covers several disciplines, but the disciplines of Communication, Design, and Marketing receive less attention than another account might provide.

A blueprint for intellectual histories of HCI was established by Ron Baecker in the opening chapters of the 1987 and 1995 editions of *Readings in Human–Computer Interaction*. It was followed in Richard Pew's chapter in the 2003 version of this handbook. Brian Shackel's (1997) account of European contributions and specialized essays by Brad Myers (1998) on HCI engineering history and Alan Blackwell (2006) on the history of metaphor in design provide further insights and references. Perlman, Green, and Wogalter (1995) is a compendium of early HCI papers that appeared in the Human Factors literature. Research on HCI within Information Systems is covered by Banker and Kaufmann (2004) and Zhang et al. (2009). Rayward (1983, 1998) and Burke (1994, 2007) review the predigital history of information science; Burke (1998) provides a focused study of an early digital effort in this field.

In recent years many popular books covering the history of personal computing have been published (e.g., Hiltzik 1999;

Bardini 2000; Hertzfeld 2005; Markoff 2005; Moggridge 2007). This introduction extends my contribution to the previous handbook. It includes new research and draws on *Timelines* columns that have appeared in *ACM Interactions* since March 2006.

Few of the aforementioned writers are trained historians. Many lived through much of the computing era as participants and witnesses, yielding rich insights and questionable objectivity. This account draws on extensive literature and hundreds of formal interviews and discussions, but everyone has biases. Personal experiences that illustrate points can enliven an account by conveying human consequences of changes that otherwise appear abstract or distant. Some readers enjoy anecdotes, whereas others find them irritating. I try to satisfy both groups by including personal examples in a short Appendix, akin to "deleted scenes" on a DVD.

Recent years have also seen the appearance of high-quality, freely accessed digital reproductions of some early works. My references include links to several such works. The reproductions do not always preserve the original pagination, but quoted passages can be found with a search tool. Finally, all prices and costs have been converted to U.S. dollars as of 2010.

Definitions: HCI, CHI, HF&E, IT, IS, LIS

The most significant term, HCI (human–computer interaction), is defined very broadly to cover major threads of research in four disciplines: (1) Human Factors/Ergonomics (HF or HF&E), (2) Information Systems (IS), (3) Computer Science (CS), and (4) Library and Information Science (LIS). The relevant literatures are difficult to explore because they differ in the use of simple terms. This is discussed later. Here I explain how several key disciplinary labels are used. CHI (Computer-Human Interaction) has a narrower focus, associated mainly with Computer Science, the Association for Computing Machinery Special Interest Group (ACM SIGCHI), and the latter's annual CHI conference. I use human factors and ergonomics interchangeably and refer to the discipline as HF&E—the Human Factors Society (HFS) became the Human Factors and Ergonomics Society (HFES) in 1992. (Some writers define ergonomics more narrowly around hardware.) Information Systems (IS) refers to the management discipline that has also been labeled Data Processing (DP) and Management Information Systems (MIS). I follow common parlance in referring to organizational information systems specialists as IT professionals or IT pros. With IS taken, I do not abbreviate Information Science. LIS (Library and Information Science) represents an old field with a new digital incarnation that includes important HCI research. Increasingly this discipline goes by simply "Information," as in newly christened Schools of Information.

HUMAN–TOOL INTERACTION AND INFORMATION PROCESSING AT THE DAWN OF THE COMPUTING ERA

In the century prior to the advent of the first digital computers, advances in technology gave rise to two fields of research that later contributed to HCI: One focused on making the human use of tools more efficient, whereas the other focused on ways to represent and distribute information more effectively.

Origin of Human Factors

Frederick Taylor (1911) employed technologies and methods developed in the late nineteenth century—photography, moving pictures, and statistical analysis—to improve work practices by reducing performance time. Time and motion studies were applied to assembly-line manufacturing and other manual tasks. Despite the uneasiness with "Taylorism" reflected in Charlie Chaplin's popular satire *Modern Times*, scientists and engineers strove to boost efficiency and productivity using this approach.

Lillian Gilbreth (1914) and her husband Frank were the first engineers to combine psychology and scientific management. Lillian Gilbreth focused more holistically than Taylor on efficiency and worker experience; she is regarded by some as the founder of modern Human Factors. Her PhD was the first awarded in industrial psychology. She went on to advise five U.S. presidents and became the first woman inducted into the National Academy of Engineering.

World War I and World War II accelerated efforts to match people to jobs, train them, and design equipment that could be more easily mastered. Engineering psychology was born during World War II after simple flaws in the design of aircraft controls (Roscoe 1997) and escape hatches (Dyson 1979) led to aircraft losses and thousands of casualties. Two legacies of World War II were respect for the potential of computing, based on its use in code breaking, and an enduring interest in behavioral requirements for design.

During the war, aviation engineers, psychologists, and physicians formed the Aeromedical Engineering Association. After the war, the terms "human engineering," "human factors," and "ergonomics" came into use, the latter primarily in Europe. For more on this history, see Roscoe (1997), Meister (1999), and HFES (2010).

Early tool use, whether by assembly-line workers or pilots, was not discretionary. If training was necessary, people were trained. One research goal was to reduce training time, but a more important goal was to increase the speed and reliability of skilled performance.

Origin of the Focus on Information

H. G. Wells, known for writing science fiction, campaigned for decades to improve society through information dissemination. In 1905, he outlined a system that might be built using another new technology of the era: index cards!

> These index cards might conceivably be transparent and so contrived as to give a photographic copy promptly whenever it was needed, and they could have an attachment into which would slip a ticket bearing the name of the locality in which the individual was last reported. A little army of attendants would be at work on this index day and night.… An incessant stream of information would come of births, of deaths, of arrivals at inns, of applications to post offices for letters,

> of tickets taken for long journeys, of criminal convictions, marriages, applications for public doles, and the like. A filter of offices would sort the stream, and all day and all night forever a swarm of clerks would go to and fro correcting this central register and photographing copies of its entries for transmission to the subordinate local stations in response to their inquiries. . . .

Would such a human-powered "Web 2.0" be a tool for social control or public information access? The image evokes the potential, and also the challenges, of the information era that is taking shape around us now, a century later.

In the late nineteenth century, technologies and practices for compressing, distributing, and organizing information bloomed. Index cards, folders, and filing cabinets—models for icons on computer displays much later—were important inventions that influenced the management of information and organizations in the early twentieth century (Yates 1989). Typewriters and carbon paper facilitated information dissemination, as did the mimeograph machine, patented by Thomas Edison. Hollerith cards and electromechanical tabulation, celebrated steps toward computing, were heavily used to process information in industry.

Photography was used to record information as well as behavior. For almost a century, microfilm was the most efficient way to compress, duplicate, and disseminate large amounts of information. Paul Otlet, Vannevar Bush, and other microfilm advocates played a major role in shaping the future of information technology.

As the cost of paper, printing, and transportation dropped in the late nineteenth and early twentieth centuries, information dissemination and the profession of librarianship grew explosively. Library associations were formed. The Dewey Decimal and Library of Congress classification systems were developed. Thousands of relatively poorly-funded public libraries sprang up to serve local demand in the United States. In Europe, government-funded libraries were established to serve scientists and other specialists in medicine and the humanities. This difference led to different approaches to technology development on either side of the Atlantic.

In the United States, library management and the training of thousands of librarians took precedence over technology development and the needs of specialists. Public libraries adopted the simple but inflexible Dewey Decimal Classification System. The pragmatic focus of libraries and emerging library schools meant that research into technology was in the province of industry. Research into indexing, cataloging, and information retrieval was variously referred to as bibliography, documentation, and documentalism.

In contrast, the well-funded European special libraries elicited sophisticated reader demands and pressure for libraries to share resources, which promoted interest in technology and information management. The Belgian Paul Otlet obtained Melvyn Dewey's permission to create an extended version of the Dewey Decimal System that supported what we would today call hypertext links. Otlet had to agree not to implement his "universal decimal classification" (UDC) in English for a time, an early example of a legal constraint on technology development. UDC is still in use in some places.

In 1926, the Carnegie Foundation dropped a bombshell: It endowed the Graduate Library School (GLS) at the University of Chicago to focus solely on research. For two decades, University of Chicago was the only university granting PhDs in library studies. GLS positioned itself in the humanities and social sciences, with research into the history of publishing, typography, and other topics (Buckland 1998). *An Introduction to Library Science*, the dominant library research textbook for 40 years, was written at Chicago (Butler 1933). *It did not mention information technology at all*. Library science was shaped by the prestigious GLS program until well into the computer era, and human–tool interaction was not among its major concerns. Documentalists, researchers who focused on technology, were concentrated in industry and government agencies.

Burke (2007, p. 15) summarized the early history with its emphasis on training librarians and other specialists: "Most information professionals … were focusing on providing information to specialists as quickly as possible. The terms used by contemporary specialists appeared to be satisfactory for many indexing tasks and there seemed no need for systems based on comprehensive and intellectually pleasing classification schemes. The goal of creating tools useful to nonspecialists was, at best, of secondary importance."

My account emphasizes when computer technologies came into what might be called "nonspecialist use." The early history of information management is significant, however, because the Web and declining digital storage costs have made it evident that everyone will soon become their own information managers, just as we are all now telephone operators. But I am getting ahead of our story. This section concludes with accounts of two individuals who, in different ways, shaped the history of information research and development.

Paul Otlet and the Mundaneum

Like his contemporary H.G. Wells, Otlet envisioned a vast network of information. But unlike Wells, Otlet and his collaborators built one. Otlet established a commercial research service around facts that he had been cataloging on index cards since the late nineteenth century. In 1919, the Belgian government financed the effort, which moved to a record center called the Mundaneum. By 1934, 15 million index cards and millions of images were organized using UDC, whose formula enabled the linking of items. Curtailed by the Depression and damaged during World War II, the work was largely forgotten. It was not cited by developers of the metaphorically identical Xerox NoteCards, an influential hypertext system of the 1980s.

Technological innovation continued in Europe with the development of mechanical systems of remarkable ingenuity (Buckland 2009). Features included the use of photoreceptors to detect light passing through holes in index cards positioned to represent different terms, enabling rapid retrieval of items on specific topics. These innovations inspired a well-known American scientist and research manager to go ahead with his endeavors.

Vannevar Bush and Microfilm Machines

Massachusetts Institute of Technology (MIT) professor Vannevar Bush was one of the most influential scientists in American history. He advised Presidents Franklin Roosevelt and Harry Truman, served as director of the Office of Scientific Research and Development, and was president of the Carnegie Institute.

Bush is remembered today for "As We May Think," his 1945 *Atlantic Monthly* essay. It described the MEMEX, a hypothetical microfilm-based electromechanical information-processing machine. The MEMEX was to be a personal workstation that enabled a professional to quickly index and retrieve documents or pictures and create hypertext-like associations among them. The essay, excerpted later in this section, inspired computer engineers and computer scientists who made major contributions to HCI in the 1960s and beyond.

Not so well known is that Bush wrote the core of his essay in the early 1930s. Then, shrouded in secrecy he spent two decades and unprecedented resources on the design and construction of several machines that comprised a subset of MEMEX features. None were successful. The details are recounted in Colin Burke's (1994) comprehensive book *Information and Secrecy: Vannevar Bush, Ultra, and the Other Memex*.

Microfilm—photographic miniaturization—had qualities that attracted Bush, as they had Otlet. Microfilm was light, could be easily transported, and was as easy to duplicate as paper records (Xerox photocopiers did not appear until 1959). The cost of handling film was brought down by technology created for the moving picture industry. Barcodelike patterns of small holes could be punched on a film and read very quickly by passing the film between light beams and photoreceptors. Microfilm was tremendously efficient as a storage medium. Memory based on relays or vacuum tubes would never be competitive, and magnetic memory, when it eventually arrived, was less versatile and far more expensive. It is easy today to overlook the compelling case that existed for basing information systems on microfilm.

Bush's machines failed because he set overly ambitious compression and speed goals, ignored patent ownership issues, and most relevant to our account, was unaware of what librarians and documentalists had learned through decades of work on classification systems. American documentalists were active, although not well funded in their work. In 1937, the American Documentation Institute (ADI) was formed, predecessor of present-day American Society for Information Science and Technology (ASIST). Had he worked with them, Bush, an electrical engineer by training, might have avoided the fatal assumption that small sets of useful indexing terms could easily be defined and agreed upon. Metadata design is still a research challenge.

At times Bush considered libraries and the public as potential users, but his machines cost far too much for library patrons to be plausible users. He began with the Federal Bureau of Investigation (FBI) in mind and focused on military uses of cryptography and information retrieval, and a major project was for the Central Intelligence Agency (CIA). Despite the classified nature of this work, through his academic and government positions, his writings, the vast resources he commandeered, and the scores of brilliant engineers he enlisted to work on microfilm projects, Bush promoted his vision and exerted influence for two decades, well into the computer era.

> Bush's vision emphasized both associative linking of information sources and discretionary use: Associative indexing, the basic idea of which is a provision whereby any item may be caused at will to select immediately and automatically another. This is the essential feature of the MEMEX.… Any item can be joined into numerous trails.… New forms of encyclopedias will appear, ready-made with a mesh of associative trails [which a user could extend].…
>
> The lawyer has at his touch the associated opinions and decisions of his whole experience and of the experience of friends and authorities. The patent attorney has on call the millions of issued patents, with familiar trails to every point of his client's interest. The physician, puzzled by a patient's reactions, strikes the trail established in studying an earlier similar case and runs rapidly through analogous case histories, with side references to the classics for the pertinent anatomy and histology. The chemist, struggling with the synthesis of an organic compound, has all the chemical literature before him in his laboratory, with trails following the analogies of compounds and side trails to their physical and chemical behavior.
>
> The historian, with a vast chronological account of a people, parallels it with a skip trail which stops only on the salient items, and can follow at any time contemporary trails which lead him all over civilization at a particular epoch. There is a new profession of trail blazers, those who find delight in the task of establishing useful trails through the enormous mass of the common record. (Bush 1945).

Bush knew that the MEMEX was not realistic. None of his many projects included designs for the "essential" associative linking. His inspirational account nicely describes present-day hands-on discretionary use of computers by professionals. But that would arrive 50 years later, built on technologies then undreamt of. Bush did not support the early use of computers, which were slow, bulky, and expensive. Computers were clearly inferior to microfilm.

1945–1955: MANAGING VACUUM TUBES

World War II changed everything. Prior to the war, government funding of research was minimal and primarily managed by the Department of Agriculture. The unprecedented investment in science and technology during the war years revealed that huge sums could be found—*for academic or industrial research that addressed national goals.* Research expectations and strategies would never again be the same.

Sophisticated electronic computation machines built before and during World War II were designed for specific purposes, such as solving equations or breaking codes. Each of the extremely expensive cryptographic machines that helped win the war was designed to attack a specific encryption device. A new one was needed whenever the enemy changed machines. These limitations spurred interest in general-purpose computational devices. Wartime improvements in technologies such as vacuum tubes made them more feasible, and their deployment brought HCI into the foreground.

When engineers and mathematicians emerged from military and government laboratories (and secret project rooms on university campuses), the public became aware of some of the breakthroughs. Development of ENIAC, arguably the first general-purpose computer, was begun in secret during the war but announced publicly as a "giant brain" only when it was completed in 1946. (Its first use, for calculations supporting hydrogen bomb development, was not publicized.) Accounts of the dimensions of ENIAC vary, but it stood 8–10-feet high, occupied about 1800 square feet, and consumed as much energy as a small town. It provided far less computation and memory than what can be acquired today for a few dollars, slipped into a pocket, and powered with a small battery.

Memory was inordinately expensive. Even the largest computers of the time had little memory, so they were used for computation and not for symbolic representation or information processing. Reducing operator burden was a key HCI focus, including replacing or resetting vacuum tubes more quickly, loading stored-program computers from tape rather than by manually attaching cables, and setting switches. Following "knobs and dials" human factors improvements, one computer operator could accomplish work that had previously required a team.

Libraries installed simple microfilm readers to assist the retrieval of information as publication of scholarly and popular material soared. Beyond that, library and library school involvement with technology was limited, even as the foundation for information science came into place. The war had forged alliances among the documentalists, electrical engineers, and mathematicians interested in communication and information management. Vannevar Bush's collaborators who were involved in this effort included Claude Shannon and Warren Weaver, coauthors in 1949 of the seminal work on information theory (called communication theory at that time). Prominent American documentalist Ralph Shaw joined Bush's efforts. Library schools continued to focus on librarianship, social science, and historical research. The GLS orientation still dominated the field. If anything the split was greater: In the 1930s, the technology-oriented ADI had included librarians and support for systems that spanned the humanities and sciences; with the coming of the war and continuing after it, ADI's concerns became those of government and Big Science.

Three Roles in Early Computing

Early computer projects employed people in the following roles: managers, programmers, and operators. Managers oversaw the design, development, and operation of projects. They specified the programs to be written and distributed the output. Scientists and engineers wrote the programs, working with mathematically adept programmers who decomposed a task into components that the computer could manage (for ENIAC, this was a team of six women). A small army of operators was needed. Once written, a program could take days to load by setting switches, dials, and cable connections. Despite innovations that boosted reliability, including operating vacuum tubes at lower power than normal and providing visible indicators of their failure, ENIAC was often stopped to locate and replace failed tubes. Vacuum tubes were reportedly wheeled around in shopping carts.

Eventually, each occupation—computer operation, management and systems analysis, and programming—became a major focus of HCI research, centered respectively in human factors, information systems, and computer science. Computers and our interaction with them evolved, but our research spectrum still reflects aspects of this early division of labor.

Grace Hopper: Liberating Computer Users

As computers became more reliable and capable, programming became a central activity. Computer languages, compilers, and constructs such as subroutines facilitated "programmer–computer interaction." Grace Hopper was a pioneer in these areas. She described her goal as freeing mathematicians to do mathematics (Hopper 1952; see also Sammet 1992). This is echoed in today's usability goal of freeing users to do their work. HCI professionals often argue that they are marginalized by software developers; in much the same way, Hopper's accomplishments have arguably been undervalued by theoretical computer scientists.

1955–1965: TRANSISTORS, NEW VISTAS

Early forecasts that the world would need few computers reflected the limitations of vacuum tubes. Solid-state computers, which first became available commercially in 1958, changed this. Computers were still used primarily for scientific and engineering tasks, but they were reliable enough not to require a staff of computer engineers. The less computer-savvy operators who oversaw them needed better interfaces. And although computers were too expensive and limited to be widely used, the potential of transistor-based computing was evident. Some researchers envisioned possibilities that were previously unimaginable.

Another major force was reaction to the then Soviet Union's launch of the Sputnik satellite in October 1957. This was a challenge to the West to invest in science and technology; becoming part of the response was a way to tie a research program to the national interest, which World War II had revealed to be so effective.

Supporting Operators: The First Systematic Human–Computer Interaction Research

> In the beginning, the computer was so costly that it had to be kept gainfully occupied for every second; people were almost slaves to feed it.
>
> **Brian Shackel (1997, p. 97)**

Almost all computer use of this period involved programs and data that were read in from cards or tape. Programs then ran without interruption until they terminated, producing printed, punched, or tape output along the way. This "batch processing" restricted human interaction to basic operation, programming, and use of the output. Of these, only computer operation, the least intellectually challenging and lowest-paying job, involved hands-on computer use.

Computer operators loaded and unloaded cards and magnetic or paper tapes, set switches, pushed buttons, read lights, loaded and burst printer paper, and put printouts into distribution bins. Operators interacted directly with the system via a teletype: Typed commands interleaved with computer responses and status messages were printed on paper that scrolled up one line at a time. Eventually, they yielded to "glass tty's" (glass teletypes), also called cathode-ray tubes (CRTs) and visual display units/terminals (VDUs/VDTs). For many years, these displays also scrolled commands and computer responses one line at a time. The price of a monochrome terminal that could display alphanumeric characters was equivalent to US$50,000 today—expensive, but only a small fraction of the cost of the computer. A large computer might have one or more consoles. Programmers did not use the interactive consoles. Programs were typically written on paper and keypunched onto cards or tape.

Improving the design of buttons, switches, and displays was a natural extension of human factors. Experts in HF&E authored the first HCI papers. In 1959 British researcher Brian Shackel published "Ergonomics for a Computer," followed in 1962 by "Ergonomics in the Design of a Large Digital Computer Console." These described console redesign for analog and digital computers called the EMIac and EMIdec 2400. Shackel (1997) described the latter as the largest computer of the time.

In the United States, American aviation psychologists created the Human Engineering Society in 1956, which was focused on skilled performance including improving efficiency, reducing errors, and training. The next year it adopted the more elegant title Human Factors Society and in 1958 it initiated the journal *Human Factors*. Sid Smith's (1963) "Man–Computer Information Transfer" marked the start of his long career with the human factors of computing.

Visions and Demonstrations

As transistors replaced vacuum tubes, a wave of imaginative writing, conceptual innovation, and prototype building swept through the research community. Some of the language is dated, notably the use of male generics, but many of the key concepts resonate even today.

J.C.R. Licklider at Bolt Beranek and Newman and Advanced Research Projects Agency

Licklider, a psychologist, played a dual role in the development of this field. He wrote influential essays and backed important research projects as a manager at Bolt Beranek and Newman (BBN) from 1957 to 1962 and as director of the Information-Processing Techniques Office (IPTO) of the Department of Defense Advanced Research Projects Agency (called ARPA and DARPA at different times) from 1962 to 1964.

BBN employed dozens of influential researchers on computer-related projects funded by the government, including John Seely Brown, Richard Pew, and many MIT faculty members such as John McCarthy, Marvin Minsky, and Licklider himself. Funding by IPTO was crucial in creating computer science departments and establishing artificial intelligence (AI) as a discipline in the 1960s. It is best known for a Licklider project that created the forerunner of the Internet called the ARPANET.

In 1960, Licklider outlined a vision he called *man–machine symbiosis*: "There are many man–machine systems. At present, however, there are no man–computer symbioses—answers are needed." The computer was "a fast information-retrieval and data-processing machine" destined to play a larger role: "One of the main aims of man–computer symbiosis is to bring the computing machine effectively into the formulative parts of technical problems" (pp. 4–5).

This required rapid, real-time interaction, which batch systems did not support. In 1962, Licklider and Wes Clark outlined the requirements of a system for "online man–computer communication." They identified capabilities that they felt were ripe for development: time-sharing of a computer among many users; electronic input–output surfaces to display and communicate symbolic and pictorial information; interactive, real-time support for programming and information processing; large-scale information storage and retrieval systems; and facilitation of human cooperation. They foresaw that other desirable technologies, such as speech recognition and natural language understanding, would be very difficult to achieve.

In a 1963 memorandum that cleverly tied computing to the emerging post-Sputnik space program, Licklider addressed his colleagues as "the members and affiliates of the Intergalactic Computer Network" and identified many features of a future Internet (Licklider 1963). His 1965 book *Libraries of the Future* expanded this vision. Licklider's role in advancing computer science and HCI is detailed by Waldrop (2001).

John McCarthy, Christopher Strachey, and Wesley Clark

McCarthy and Strachey worked out details of time-sharing, which made interactive computing possible (Fano and Corbato 1966). Apart from a few researchers who had access to computers built with no-expenses-spared military funding, computer use was too expensive to support exclusive individual access. Time-sharing allowed several (and later dozens) simultaneous users to work at terminals. Languages were developed to facilitate the control and programming of time-sharing systems (e.g., JOSS in 1964).

Clark was instrumental in building the TX-0 and TX-2 at MIT's Lincoln Laboratory to demonstrate time-sharing and other innovative concepts. These machines, which cost on the order of US$10 million, helped establish the Boston area as a center for computer research. The TX-2 was the most powerful and capable computer in the world at the time. It was much less powerful and capable than a present-day smartphone. Clark and Ivan Sutherland discussed this era in a CHI'05 panel, which is accessible online (Buxton 2006).

Ivan Sutherland and Computer Graphics

Sutherland's 1963 PhD thesis may be the most influential document in the history of HCI. His Sketchpad system, built on TX-2 to make computers "more approachable," launched computer graphics, which would have a decisive impact on HCI 20 years later. A nice version restored by Alan Blackwell and Kerry Rodden is available (http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-574.pdf).

Sutherland demonstrated iconic representations of software constraints, object-oriented programming concepts, and the copying, moving, and deleting of hierarchically organized objects. He explored novel interaction techniques, such as picture construction using a light pen. He facilitated visualization by separating the coordinate system used to define a picture from the one used to display it, and demonstrated animated graphics, noting the potential for digitally rendered cartoons 20 years before *Toy Story*. His frank descriptions enabled others to make rapid progress in the field—when engineers found Sketchpad too limited for computer-assisted design (CAD), he called the trial a "big flop" and indicated why.

In 1964, with his PhD behind him, Sutherland succeeded Licklider as the director of IPTO. Among those he funded was Douglas Engelbart at the Stanford Research Institute (SRI).

Douglas Engelbart: Augmenting Human Intellect

In 1962, Engelbart published "Augmenting Human Intellect: A Conceptual Framework." Over the next several years he built systems that made astonishing strides toward realizing this vision. He also supported and inspired engineers and programmers who went on to make major independent contributions.

Echoing Bush and Licklider, Engelbart saw the potential for computers to become congenial tools that people would choose to use interactively:

> By 'augmenting human intellect' we mean increasing the capability of a man to approach a complex problem situation, to gain comprehension to suit his particular needs, and to derive solutions to problems.... By 'complex situations' we include the professional problems of diplomats, executives, social scientists, life scientists, physical scientists, attorneys, designers.... We refer to a way of life in an integrated domain where hunches, cut-and-try, intangibles, and the human 'feel for a situation' usefully coexist with powerful concepts, streamlined terminology and notation, sophisticated methods, and high-powered electronic aids.
>
> **(Engelbart 1962, p. 1)**

Engelbart used ARPA funding to rapidly develop and integrate an extraordinary set of prototype applications into his NLS system. In doing so, he conceptualized and implemented the foundations of word processing, invented or refined input devices including the mouse and the multikey control box, and made use of multidisplay environments that integrated text, graphics, and video in windows. These unparalleled advances were demonstrated in a sensational 90-minute live event at the 1968 Fall Joint Computer Conference in San Francisco, California (http://sloan.stanford.edu/MouseSite/1968Demo.html). The focal point for interactive systems research in the United States was moving from the East Coast to the West Coast.

Engelbart, an engineer, supported human factors testing to improve efficiency and reduce errors in skilled use, focusing on effects of fatigue and stress. Engelbart's systems required training. He felt that people should be willing to tackle a difficult interface if it delivered great power once mastered. Unfortunately, the lack of concern for initial usability was a factor in Engelbart's loss of funding. His demonstration became something of a success disaster: DARPA was impressed and installed NLS, but found it too difficult to use (Bardini 2000). Years later, the question "Is it more important to optimize for skilled use or initial use?" was widely debated, and still occasionally surfaces in HCI discussions.

Ted Nelson's Vision of Interconnectedness

In 1960, Ted Nelson, a graduate student in sociology who coined the term hypertext, founded Project Xanadu. The goal was an easily used computer network. In 1965, he published a paper titled "A File Structure for the Complex, the Changing and the Indeterminate." Nelson continued to write stirring calls for systems to democratize computing through a highly interconnected, extensible network of digital objects (e.g., Nelson 1973). Xanadu was never fully realized. Nelson did not consider the early World Wide Web to be an adequate realization of his vision, but lightweight technologies such as weblogs, wikis, collaborative tagging, and search enable many of the activities he envisioned.

Later, Nelson (1996) foresaw intellectual property issues arising in digital domains and coined the term "micropayment." Although his solutions were again not fully implemented, they drew attention to important issues.

From Documentation to Information Science

The late 1950s saw the last major investments in microfilm and other predigital systems. The most ambitious were military and intelligence systems, including Vannevar Bush's final efforts (Burke 1994). Documentalists began to see that declining memory costs would enable computation engines to become information-processing machines. The conceptual evolution was relatively continuous, but at the institutional level change could come swiftly. New professions—mathematicians and engineers—were engaged in technology development, new initiatives were launched that still bore few ties to contemporary librarianship or the humanities orientation of library schools. A new banner was needed.

Merriam Webster dates the term information science to 1960. Conferences held at Georgia Institute of Technology in 1961 are credited with shifting the focus from information as a technology to information as an incipient science. In 1963, chemist-turned-documentalist Jason Farradane taught the first information science courses at City University, London, United Kingdom. The profession of chemistry had long invested in organizing its literature systematically, and another chemist-turned-documentalist Allen Kent was at the center of a major information science initiative at the University of Pittsburgh (Aspray 1999). In the early 1960s, Anthony Debons, a psychologist and friend of Licklider, organized a series of NATO-sponsored congresses at Pittsburgh. Guided by Douglas Engelbart, these meetings centered on people and on how technology could augment their activities. In 1964 the Graduate Library School at the University of Pittsburgh became the Graduate School of Library and Information Sciences, and Georgia Tech formed a School of Information Science initially with one full-time faculty member.

Conclusion: Visions, Demos, and Widespread Use

Progress in HCI can be understood in terms of inspiring visions, conceptual advances that enable aspects of the visions to be demonstrated in working prototypes, and the evolution of design and application. The engine, enabling visions to be realized and soon thereafter to be widely deployed, was the relentless hardware advance that produced devices that were millions of times more powerful than the much more expensive systems designed and used by the pioneers.

At the conceptual level, much of the basic foundation for today's graphical user interfaces (GUIs) was in place by 1965. However, at that time it required individual use of a US$10-million custom-built machine. Pew (2003, p. 3) describes the 1960 Digital Equipment Corporation (DEC) PDP-1 as a breakthrough, "truly a computer with which an individual could interact." The PDP-1 came with a CRT display, keyboard, light pen, and paper tape reader. It cost about US$1 million and had the capacity that a Radio Shack TRS 80 had 20 years later. It required considerable technical and programming support. Even the PDP-1 could only be used by a few fortunate researchers.

Licklider's man–computer symbiosis, Engelbart's augmenting human intellect, and Nelson's "conceptual framework for man–machine everything" described a world that did not exist. It was a world in which attorneys, doctors, chemists, and designers chose to become hands-on users of computers. For some time to come, the reality would be that most hands-on users were computer operators engaged in routine, nondiscretionary tasks. As for the visions, 40 years later some of the capabilities are taken for granted, some are just being realized, and others remain elusive.

1965–1980: HUMAN–COMPUTER INTERACTION PRIOR TO PERSONAL COMPUTING

Control Data Corporation launched the transistor-based 6000 series computer in 1964. In 1965, commercial computers based on integrated circuits arrived with the IBM System/360. These powerful systems, later called mainframes to distinguish them from minicomputers, firmly established computing in the business realm. Each of the three computing roles—operation, management, and programming—became a significant profession.

Operators still interacted directly with computers for routine maintenance and operation, and as time-sharing developed, hands-on use expanded to include data entry and other repetitive tasks. Managers and systems analysts oversaw hardware acquisition, software development, operation, and the use of output. They were usually not hands-on users, although people who relied on printed output and reports did call themselves "computer users."

Apart from those working in research settings, few programmers were direct users until late in this period. Many prepared flowcharts and wrote programs on paper forms. Keypunch operators then punched the program instructions onto cards, which were sent to computer centers for computer operators to load into the computer and run. Printouts and other output were picked up later. Many programmers used computers directly when they could, but the cost generally dictated more efficient division of labor.

We are focusing on broad trends. Business computing took off in the mid-1960s, although the 1951 LEO I was probably the first commercial business computer. This interesting venture, which ended with the arrival of the mainframe era, is detailed in Wikipedia (under 'LEO computer') and the books and articles referenced there.

Human Factors and Ergonomics Embrace Computer Operation

In 1970, Brian Shackel founded the Human Sciences and Advanced Technology (HUSAT) center at Loughborough University in Leicestershire, the United Kingdom, which is devoted to ergonomics research that emphasizes HCI. Sid Smith and other human factors engineers worked on input and output issues, such as the representation of information on displays (e.g., Smith, Farquhar, and Thomas 1965) and computer-generated speech (Smith and Goodwin 1970). The Computer Systems Technical Group (CSTG) of the HFS was formed in 1972, and soon it was the largest technical group in the society.

The general *Human Factors* journal was joined in 1969 by the computer-focused *International Journal of Man–Machine Studies (IJMMS)*. The first widely read HCI book was James Martin's (1973) *Design of Man–Computer Dialogues*. Martin's comprehensive survey of interfaces for operation and data entry began with an arresting opening chapter that described a world in transition. Extrapolating from declining hardware prices, he wrote, "The terminal or console operator, instead of being a peripheral consideration, will become the tail that wags the whole dog.… The computer industry will be forced to become increasingly concerned with the usage of people, rather than with the computer's intestines" (pp. 3–4).

In the mid-1970s, U.S. government agencies responsible for agriculture and social security initiated large-scale data-processing system projects, described by Pew (2003). Although not successful, these efforts led to methodological innovations in the use of style guides, usability laboratories, prototyping, and task analysis.

In 1980, three significant HF&E books were published: two on VDT design (Cakir, Hart, and Stewart 1980; Grandjean and Vigliani 1980) and one general guideline (Damodaran, Simpson, and Wilson 1980). Drafts of a German work on VDT standards, made public in 1981, provided an economic incentive to design for human capabilities by threatening to ban noncompliant products. Later in the same year, a corresponding American National Standards Institute standards group for "office and text systems" was formed.

Information Systems (IS) Addresses the Management of Computing

Companies acquired expensive business computers to address major organizational concerns. Even when the principal concern was simply to appear modern (Greenbaum 1979), the desire to show benefits from a multimillion dollar investment could chain managers to a computer almost as tightly as were the operator and data entry "slaves." In addition to being expected to make use of output, they might encounter resistance to system acceptance.

Beginning in 1967, the journal *Management Science* published a column titled "Information Systems in Management Science." Early definitions of IS included "an integrated man–machine system for providing information to support the operation, management, and decision-making functions in an organization" (Davis 1974) and "the effective design, delivery, and use of information systems in organizations" (Keen 1980 quoted in Zhang, Nah, and Preece 2004). In 1968, an MIS center and degree program was established at Minnesota. It initiated several influential research streams and in 1977 launched *MIS Quarterly*, the leading journal in the field. The MIS field juxtaposed a focus on specific tasks in organizational settings with demands for general theory and precise measurement, a challenging combination.

A historical survey (Banker and Kaufmann 2004) identifies HCI as one of five major IS research streams and dates it back to Ackoff's (1967) paper describing challenges in handling computer-generated information. There was some research into hands-on operator issues such as data entry and error messages, but for a decade most HCI work in IS dealt with the users of information, typically managers. Research included the design of printed reports, but the drive for theory led to a strong focus on cognitive styles: individual differences in how people (notably managers) perceive and process information. Articles on HCI were published in the human factors-oriented *IJMMS* as well as management journals.

Sociotechnical approaches to system design (Mumford 1971, 1976; Bjørn-Andersen and Hedberg 1977) were developed in response to user difficulties and resistance. These involved educating representative workers about technological possibilities and involving them in design, in part to increase their acceptance of the resulting system. Late in this period, sophisticated views of the complex social and organizational dynamics around system adoption and use emerged (e.g., Kling 1980; Markus 1983).

Programming: Subject of Study, Source of Change

Even programmers who were not hands-on users were interacting with computers, and more than 1000 research papers on variables affecting programming performance were published in the 1960s and 1970s (Baecker and Buxton 1987). Most were studies of the behavior of programmers in isolation, independent of organizational context. Influential reviews of this work included Gerald Weinberg's landmark *The Psychology of Computer Programming* in 1971; Ben Shneiderman's *Software Psychology: Human Factors in Computer and Information Systems* in 1980; and Beau Sheil's 1981 review of studies of programming notation (conditionals, control flow, data types), practices (flowcharting, indenting, variable naming, commenting), and tasks (learning, coding, debugging).

Software developers changed the field through invention. In 1970, Xerox Palo Alto Research Center (PARC) was founded to advance computer technology by developing new hardware, programming languages, and programming environments. It attracted researchers and system builders from the laboratories of Engelbart and Sutherland. In 1971, Allen Newell of Carnegie Mellon University (CMU), Pennsylvania, proposed a project to PARC, which was launched 3 years later: "Central to the activities of computing—programming, debugging, etc.—are tasks that appear to be within the scope of this emerging theory [a psychology of cognitive behavior]" (Card and Moran 1986, p. 183).

Like HUSAT, which was also launched in 1970, PARC had a broad charter. HUSAT focused on ergonomics, anchored in the tradition of nondiscretionary use, one component of which was the human factors of computing. PARC focused on computing, anchored in visions of discretionary use, one component of which was also the human factors of computing. Researchers at PARC, influenced by cognitive psychology, extended the primarily perceptual motor focus of human factors to higher-level cognition, whereas HUSAT, influenced by sociotechnical design, extended human factors by considering organizational factors.

Computer Science: A New Discipline

Computer science departments in educational institutions emerged in the mid-1960s. ome originated in engineering, others in applied mathematics. From engineering, computer graphics was a specialization of particular relevance to HCI. Applied mathematics was the background of many early AI researchers, which has interacted with HCI in complex ways in subsequent years.

The expensive early machines capable of interesting work were funded without consideration to cost by branches of the military. Technical success was the sole evaluation criterion (Norberg and O'Neill 1996). Directed by Licklider, Sutherland, and their successors, ARPA played a major role. The need for heavy funding concentrated researchers in a few centers, which bore little resemblance to the batch and time-shared business computing environments of that era.

User needs differed: The technically savvy hands-on users in research settings did not press for low-level interface enhancements.

The computer graphics and AI perspectives that arose in these centers differed from the perspectives of HCI researchers who focused on less expensive, more widely deployed systems. Computer graphics and AI required processing power; hardware advances meant declining cost for the same high level of computation. For HCI researchers, hardware advances meant greater computing capability at the same low price. Only later would this difference diminish, when widely available machines could support graphical interfaces and some AI programs. Despite this gap, between 1965 and 1980 some computer science researchers focused on interaction, which is not surprising given that interaction was an element of the visions formulated in the previous decade.

Computer Graphics: Realism and Interaction

In 1968, Sutherland joined David Evans to establish an influential computer graphics laboratory at the University of Utah. The Utah Computer Science Department was founded in 1965, as part of computer science's first move into academic prominence. Utah contributed to the western migration as graduates of the laboratory, including Alan Kay and William Newman (and later Jim Blinn and Jim Clark), went to California. Most graphics systems at the time were built on the DEC PDP-1 and PDP-7. These expensive machines—the list price of a high-resolution display alone was equivalent to more than US$100,000 in today's dollars—were in principle capable of multitasking, but in practice most graphics programs required all of a processor's cycles.

In 1973 the Xerox Alto arrived, a powerful step toward realizing Alan Kay's vision of computation as a medium for personal computing (Kay and Goldberg 1977). The Alto was too expensive to be widely used—it was never widely marketed—and not powerful enough to support high-end graphics research, but it did support graphical interfaces of the kind Engelbart had prototyped. In doing so, the Alto signaled the approach of inexpensive, interactive, personal machines capable of supporting graphics. Computer graphics researchers had to decide whether to focus on high-end graphics or on more primitive features that would soon run on widely affordable machines.

William Newman, coauthor in 1973 of the influential *Principles of Interactive Computer Graphics*, described the shift in a personal communication: "Everything changed—the computer graphics community got interested in realism; I remained interested in interaction, and I eventually found myself doing HCI." He was not alone. Other graphics researchers whose focus shifted to broader interaction issues included Ron Baecker and Jim Foley. Foley and Wallace (1974, p. 462) identified requirements for designing "interactive graphics systems whose aim is good symbiosis between man and machine." The shift was gradual: A total of 18 papers in the first SIGGRAPH conference, in 1974, had the words "interactive" or "interaction" in their titles. A decade later, there would be none.

At Xerox, Larry Tesler and Tim Mott recognized that Alto could support a graphical interface accessible to untrained people. The latter point had not been important given the prior focus on trained, expert performance. By early 1974, Tesler and Mott had developed the Gypsy text editor. Gypsy and Xerox's Bravo editor developed by Charles Simonyi preceded and influenced Microsoft Word (Hiltzik 1999).

The focus on interaction was highlighted in 1976 when SIGGRAPH sponsored a 2-day workshop in Pittsburgh, User-Oriented Design of Interactive Graphics Systems (UODIGS). Participants who were later active in CHI included Jim Foley, William Newman, Ron Baecker, John Bennett, Phyllis Reisner, and Tom Moran. Licklider and Nicholas Negroponte presented vision papers. The conference was managed by the chair of Pittsburgh's computer science department. One participant was Anthony Debons, Licklider's friend who had helped build Pittsburgh's world-renowned information science program. The UODIGS'76 workshop arguably marked the end of a visionary period, embodying an idea whose time had not quite yet come. Licklider saw it clearly:

> Interactive computer graphics appears likely to be one of the main forces that will bring computers directly into the lives of very large numbers of people during the next two or three decades. Truly user-oriented graphics of sufficient power to be useful to large numbers of people has not been widely affordable, but it will soon become so and, when it does, the appropriateness and quality of the products offered will to a large extent determine the future of computers as intellectual aids and partners of people.
>
> **(Licklider 1976, p. 89)**

UODIGS was not repeated. Despite the stature of its participants, the 150-page proceedings were not cited. Not until 1981 was another user-oriented design conference held, after which such conferences were held every year. Application of graphics was not quite at hand; most HCI research remained focused on interaction driven by commands, forms, and full-page menus.

Artificial Intelligence: Winter Follows Summer

In the late 1960s and early 1970s AI burst onto the scene, promising to transform HCI. It did not go as planned. Logically, AI and HCI are closely related. What are intelligent machines for if not to interact with people? Research on AI has influenced HCI: Speech recognition and natural language are perennial HCI topics; expert, knowledge-based, adaptive, and mixed-initiative systems have been tried, as have applications of production systems, neural networks, and fuzzy logic. Today, human–robot interaction and machine learning are attracting much attention.

Although some AI features make it into systems and applications, frequent predictions that powerful machines would soon bring major AI technologies into wide use and thus become a focus of HCI research were not borne out. AI did not come into focus in HCI, and AI researchers showed limited interest in HCI.

To piece this together one requires a brief review of early AI history. The term "artificial intelligence" first appeared in a 1955 call by John McCarthy for a meeting on machine intelligence that was held in Dartmouth. In 1956, Alan Turing's prescient essay "Computing Machinery and Intelligence" attracted attention when it was reprinted in *The World of Mathematics*. (It was first published in 1950, as were Claude Shannon's "Programming a Computer for Playing Chess" and Isaac Asimov's *I, Robot*, which explored his three laws of robotics.) Newell and Simon presented their logic theory machine in 1956 and then focused on developing a general problem solver. McCarthy invented the LISP programming language in 1958 (McCarthy 1960).

Many AI pioneers were trained in mathematics and logic, where almost everything can be derived from a few axioms and a small set of rules. Mathematical ability is considered a high form of intelligence, even by non-mathematicians. AI researchers anticipated that machines that operate logically and tirelessly would achieve high levels of intelligence—applying a small set of rules to a limited number of objects. Early AI focused on theorem-proving and games and problems that had a strong logical focus, such as chess and go. McCarthy (1988), who espoused predicate calculus as a foundation for AI, summed it up as follows:

> As suggested by the term 'artificial intelligence', we were not considering human behavior except as a clue to possible effective ways of doing tasks. The only participants who studied human behavior were Newell and Simon. (The goal) was to get away from studying human behavior and consider the computer as a tool for solving certain classes of problems. Thus, AI was created as a branch of computer science and not as a branch of psychology.

Unfortunately, by ignoring psychology, mathematicians overlooked the complexity and inconsistency that mark human beings and our social constructs. Underestimating the complexity of intelligence, they overestimated the prospects for creating it artificially. Hyperbolic predictions and AI have been close companions. In the summer of 1949 the British logician and code breaker Alan Turing wrote in the *London Times*:

> I do not see why [the computer] should not enter any one of the fields normally covered by the human intellect, and eventually compete on equal terms. I do not think you can even draw the line about sonnets, though the comparison is perhaps a little bit unfair because a sonnet written by a machine will be better appreciated by another machine.

Optimistic forecasts by the 1956 Dartmouth workshop participants attracted considerable attention. When they collided with reality, a pattern was established that was to play out repeatedly. Hans Moravec (1998) wrote:

> In the 1950s, the pioneers of AI viewed computers as locomotives of thought, which might outperform humans in higher mental work as prodigiously as they outperformed them in arithmetic, if they were harnessed to the right programs.… By 1960 the unspectacular performance of the first reasoning and translation programs had taken the bloom off the rose.

A significant part of the pattern is that HCI thrives on resources that are freed when interest in AI declines. In 1960, with the bloom wearing off the AI rose, the managers of MIT's Lincoln Laboratory looked for new uses for the massive government-funded TX-0 and TX-2 computers. Ivan Sutherland's Sketchpad and early computer graphics were a result.

The response to Sputnik reversed the downturn in AI prospects. Licklider, as director of ARPA's IPTO (1962–1964), provided extensive support for computer science in general and AI in particular. MIT's Project Mac, founded in 1963 by Marvin Minsky and others, initially received US$13 million per year, rising to US$24 million in 1969. ARPA sponsored the AI Laboratory at SRI, AI research at CMU, and Nicholas Negroponte's Machine Architecture Group at MIT. A dramatic early achievement, SRI's Shakey the Robot, was featured in articles in *Life* (Darrach 1970) and *National Geographic* (White 1970). Given a simple but nontrivial task, Shakey could apparently go to the desired location, scan and reason about the surroundings, and move objects as needed to accomplish the goal (for Shakey at work, see http://www.ai.sri.com/shakey/).

In 1970, Negroponte outlined a case for machine intelligence: "Why ask a machine to learn, to understand, to associate courses with goals, to be self-improving, to be ethical—in short, to be intelligent?" He noted common reservations, "People generally distrust the concept of machines that approach (and thus why not pass?) our own human intelligence," and identified a key problem: "Any design procedure, set of rules, or truism is tenuous, if not subversive, when used out of context or regardless of context." This insight, that it is risky to apply algorithms without understanding the situation at hand, led Negroponte to a false inference: "*It follows that a mechanism must recognize and understand the context before carrying out an operation*." (Negroponte 1970, p. 1; my italics).

A perfectly reasonable alternative is that the mechanism is guided by humans who understand the context: Licklider's human–machine symbiosis. Overlooking this, Negroponte built a case for an ambitious research program:

> Therefore, a machine must be able to discern changes in meaning brought about by changes in context, hence, be intelligent. And to do this, it must have a sophisticated set of sensors, effectors, and processors to view the real world directly and indirectly.… A paradigm for fruitful conversations must be machines that can speak and respond to a natural language.… But, the tete-à-tete [*sic*] must be even more direct and fluid; it is gestures, smiles, and frowns that turn a conversation into a dialogue.… Hand waving often carries as much meaning as text. Manner carries cultural information: The Arabs use their noses, the Japanese nod their heads.… Imagine a machine that can follow your design methodology and at the same time discern and assimilate your conversational idiosyncrasies. This same machine after observing

> your behavior could build a predictive model of your conversational performance. Such a machine could then reinforce the dialogue by using the predictive model to respond to you in a manner that is in rhythm with your personal behavior and conversational idiosyncrasies.… The dialogue would be so intimate—even exclusive—that only mutual persuasion and compromise would bring about ideas, ideas unrealizable by either conversant alone. No doubt in such a symbiosis it would not be solely the human designer who would decide when the machine is relevant (pp. 1–13).

The same year, Negroponte's MIT colleague Minsky went further, as reported in *Life*:

> In from three to eight years we will have a machine with the general intelligence of an average human being. I mean a machine that will be able to read Shakespeare, grease a car, play office politics, tell a joke, and have a fight. At that point, the machine will begin to educate itself with fantastic speed. In a few months, it will be at genius level and a few months after that its powers will be incalculable.
>
> **(Darrach 1970, p. 60)**

Other AI researchers told Darrach that Minsky's timetable was ambitious: "Give us 15 years was a common remark—but all agreed that there would be such a machine and that it would precipitate the third Industrial Revolution; wipe out war and poverty; and roll up centuries of growth in science, education, and the arts" (Darrach 1970, p. 60).

Such predictions were common. In 1960, Nobel laureate and AI pioneer Herb Simon wrote: "Machines will be capable, within 20 years, of doing any work that a man can do." (Simon 1960, p. 38). Five years later, I. J. Good, an Oxford mathematician, wrote, "The survival of man depends on the early construction of an ultraintelligent machine" that "could design even better machines; there would then unquestionably be an 'intelligence explosion', and the intelligence of man would be left far behind" (Good 1965, pp. 31–33).

The Darrach article ended by quoting Ross Quillian:

> I hope that man and these ultimate machines will be able to collaborate without conflict. But if they can't, we may be forced to choose sides. And if it comes to choice, I know what mine will be. My loyalties go to intelligent life, no matter in what medium it may arise".
>
> **(Darrach 1970, p. 68)**

It is important to understand the anxieties of the time and the consequences of such claims. The world had barely avoided a devastating thermonuclear war during the Cuban missile crisis of 1962. Leaders seemed powerless to defuse the Cold War. Responding to a sense of urgency, ARPA initiated major programs in speech recognition and natural language understanding in 1971.

Ironically, central to funding this research was a psychologist not wholly convinced by the vision. Citing an Air Force study that predicted that intelligent machines might take 20 years to arrive, Licklider (1960) noted that in this interval HCI would be useful: "That would leave, say, 5 years to develop man–computer symbiosis and 15 years to use it. The 15 may be 10 or 500, but those years should be intellectually the most creative and exciting in the history of mankind." Ten to five hundred years represent breathtaking uncertainty. Recipients of Licklider's funding were on the optimistic end of this spectrum.

Five years later, disappointed with the progress, ARPA discontinued speech and language support—for a while. In Europe, a similar story unfolded. Through the 1960s, AI research expanded in Great Britain. A principal proponent was Turing's former colleague Donald Michie. Then in 1973 the Lighthill report, commissioned by the Science and Engineering Research Council, reached generally negative conclusions about AI's prospects for scaling up to address real-world problems. Almost all government funding was cut off.

The next decade was an AI winter, a recurring season in which research funding is withheld due to disillusionment over unfulfilled promises. The bloom was again off the rose, but it would prove to be a hardy perennial (Grudin 2009).

Library Schools Embrace Information Science

Early information science research and studies of "human information behavior" were initiated in the 1960s and 1970s, which focused on scholarship and application in science and engineering (Fidel 2011). The response to Sputnik proved that Big Science research did not end when the war ended. Aligning their work with national priorities became a priority for many researchers.

The terms "information science,""information technology," and "information explosion" swept into use. The Pittsburgh and Georgia Tech programs flourished. Pittsburgh created the first information science PhD program in the United States in 1970, identifying humans "as the central factor in the development of an understanding of information phenomena" (Aspray 1999, p. 12). The program balanced behavioral sciences (psychology, linguistics, communication) and technical grounding (automata theory, computer science). In 1973, Pittsburgh established the first information science department. Its program developed a strong international reputation. Slowly, the emphasis shifted from behavior to technology. On being awarded a major National Science Foundation (NSF) center grant in 1966, the Georgia Tech school expanded. In 1970 it became a PhD-granting school, rechristened as Information and Computer Science.

In 1968, the American Documentation Institute became the American Society for Information Science, and 2 years later the journal *American Documentation* became *Journal of the American Society for Information Science*. In 1978, the ACM Special Interest Group on Information Retrieval (SIGIR) was formed. It launched an annual conference for "Information Storage and Retrieval" (since 1982, "Information Retrieval"), modeled on a 1971 conference. In 1984, the American Library Association belatedly embraced the i-word by creating the Association for Library and Information Science Education (ALISE), which convened an annual research conference.

By 1980, schools at over a dozen universities had added the word information to their titles. Many were library school transitions. Delivery on the promise of transformative technology lagged, however. For example, from 1965 to 1972 the Ford and Carnegie Foundations, NSF, DARPA, and the American Newspaper Publishers Association invested over US$30 million in MIT's Project Intrex (Burke 1998). The largest nonmilitary information research project of its time, Intrex was to be the library of the future. Online catalogs were to include up to 50 index fields per item, accessible on CRT displays, with full text of books and articles converted to microfilm and read via television displays. None of this proved feasible.

Terminal-based computing costs declined. The ARPANET debuted in 1969, and supported e-mail in 1971 and file sharing in 1973. This spurred visions of a "network society" of the future (Hiltz and Turoff 1978).

As an aside, the technological optimism that marked this era lacked the nuanced psychological insight of E. M. Forster who in 1909 anticipated AI and networking developments in his remarkable story *The Machine Stops*.

1980–1985: DISCRETIONARY USE COMES INTO FOCUS

In 1980, most HF&E and IS research focused on the down-to-earth business of making efficient use of expensive mainframes. The beginning of a major shift went almost unnoticed. Less expensive but highly capable minicomputers based on LSI technology enabled DEC, Wang Laboratories, and Data General to make inroads into the mainframe market. At the low end, home computers gained traction. Students and hobbyists were drawn to these minis and micros, creating a population of hands-on discretionary users. There were experimental trials of online library catalogs and electronic journals.

Then, between 1981 and 1984 a flood of innovative and powerful computers were released: Xerox Star; IBM PC; Apple Lisa; LISP machines from Symbolics and Lisp Machines, Inc. (LMI); workstations from Sun Microsystems and Silicon Graphics; and the Apple Macintosh. On January 1, 1984, AT&T's breakup into competing companies took effect. AT&T had more employees and more customers than any other U.S. company. It was a monopoly: Neither its customers nor its employees had discretion in technology use. Both AT&T and its Bell Laboratories research division had employed human factors research to improve training and increase efficiency. Suddenly freed from a ban on entering the computer business, AT&T launched the ill-fated Unix PC in 1985. AT&T and the new regional operating companies now faced customers who had choices, and their HCI focus broadened accordingly (Israelski and Lund 2003).

In general, lower-priced computers created markets for shrink-wrap software. For the first time, computer and software companies targeted significant numbers of nontechnical hands-on users who received little or no formal training. It had taken 20 years, but early visions were being realized. Nonprogrammers were choosing to use computers to do their work. The psychology of discretionary users intrigued two groups: (1) psychologists who liked to use computers and (2) technology companies who wanted to sell to discretionary users. Not surprisingly, computer and telecommunication companies started hiring a lot of experimental psychologists.

Discretion in Computer Use

Technology use lies on a continuum bracketed by the assembly-line nightmare of *Modern Times* and the utopian vision of completely empowered individuals. To use a technology or not to use it—sometimes we have a choice, other times we do not. On the phone, we may have to wrestle with speech recognition and routing systems. At home, computer use may be largely discretionary. The workplace often lies in between: Technologies are prescribed or proscribed, but we ignore some injunctions or obtain exceptions, we use some features but not others, and we join with colleagues to press for changes.

For early computer builders, work was more a calling than a job, but operation required a staff to carry out essential if less interesting tasks. For the first half of the computing era, most hands-on use was by people with a mandate. Hardware innovation, more versatile software, and steady progress in understanding the psychology of users and tasks—and transferring that understanding to software developers—led to hands-on users who had more choice regarding how they worked. Rising expectations played a role; people learned that software is flexible and expected it to be more congenial. Competition among vendors produced alternatives. With more emphasis on marketing to consumers came more emphasis on user-friendliness.

Discretion is not all-or-none. No one must use a computer, but many jobs and pastimes require it. People can resist, sabotage, or quit their jobs. However, a clerk or a systems administrator has less discretion than someone using technology for a leisure activity. For an airline reservation clerk, computer use is mandatory. For a traveler booking a flight, computer use is discretionary. This distinction, and the shift toward greater discretion, is at the heart of the history of HCI.

The shift was gradual. About 30 years ago, John Bennett (1979) predicted that discretionary use would lead to more emphasis on usability. The 1980 book *Human Interaction with Computers*, edited by Harold Smith and Thomas Green, perched on the cusp. It included an article by Jens Rasmussen, "The Human As a Systems Component," that covered the nondiscretionary perspective. One-third of the book covered research on programming. The remainder addressed "nonspecialist people," discretionary users who are not computer savvy. Smith and Green wrote, "It is not enough just to establish what computer systems can and cannot do; we need to spend just as much effort establishing what people can *and want to do*" (p. viii, italics in original).

A decade later, Liam Bannon (1991) noted broader implications of a shift "from human factors to human actors." The trajectory is not always toward choice. Discretion can be curtailed—for example, word processor use is now often a job requirement and not an alternative to using a typewriter. Even in an era of specialization, customization, and competition, the exercise of choice varies over time and across contexts. Discretion is only one factor, but an analysis of its role casts light on how HCI efforts differ and why they have remained distinct through the years.

Minicomputers and Office Automation

Cabinet-sized minicomputers that could support several people were available from the mid-1960s. By late 1970s, superminis such as the VAX 11/780 supported integrated suites of productivity tools. In 1980, DEC, Data General, and Wang Laboratories were growth companies near Boston.

A minicomputer could handle personal productivity tools or a database of moderate size. Users sat at terminals. With "dumb terminals," the central processor handled each keystroke. Other terminals had a processor that supported a user who entered a screenful of data, which was then on command sent as a batch to the central processor. These minis could provide a small group (or office) with file-sharing, word-processing, spreadsheet, and e-mail, and manage output devices. They were marketed as "office systems," "office automation (OA) systems," or "office information systems" (OIS).

The 1980 Stanford International Symposium on Office Automation marked the emergence of a research field that remained influential for a decade and then faded away. Douglas Engelbart contributed two papers to the proceedings of this symposium (Landau, Bair, and Siegman 1982). In the same year, the American Federation of Information-Processing Societies (AFIPS, the parent organization of ACM and Institute of Electrical and Electronics Engineers [IEEE] at the time) held the first of seven annual OA conferences and product exhibitions. Also in 1980, ACM formed the Special Interest Group on Office Automation (SIGOA), which launched the biennial Conference on Office Information Systems (COIS) 2 years later. In 1983, the journal *ACM Transactions on Office Information Systems (TOOIS)* emerged, which was 1 year after the emergence of the independent journal *Office: Technology and People*.

You might ask "what is all this with offices?" Minicomputers brought down the price of computers to fit into the budget of a small workgroup or an office. (The attentive reader will anticipate: The personal computer era is approaching.) Office Information Systems, which focused on the use of minicomputers, was positioned alongside MIS, which focused on mainframes. Its scope was reflected in the charter of *TOOIS*: database theory, AI, behavioral studies, organizational theory, and communications. Minis were accessible to database researchers. Digital's PDP series was a favorite with AI researchers until LISP machines flourished. Minis were familiar to behavioral researchers who used them to run and analyze psychology experiments. Computer-mediated communication (CMC) was an intriguing new capability: Networking was still rare, but people at different terminals of a minicomputer could exchange e-mail or chat in real time. Minis became interactive computers of choice for many organizations. As a consequence, Digital became the second largest computer company in the world and Dr. Wang the fourth wealthiest American.

Researchers were discretionary users, but few office workers chose their tools. The term "automation" was challenging and exciting to researchers, but it conjured up less pleasant images for office workers. Some researchers, too, preferred Engelbart's focus on augmentation rather than automation.

Papers in the SIGOA newsletter, COIS, and *TOOIS* included technical work on database theory, a modest number of AI papers (the AI winter had not yet ended), decision support and CMC papers from the IS community, and behavioral studies by researchers who later joined CHI. Papers on information systems were prevalent in the newsletter and technical papers in *TOOIS,* which also published numerous behavioral studies until the journal *Human–Computer Interaction* started in 1985.

Although OA/OIS research was eventually absorbed by other fields, it identified and called attention to important emerging topics, including hypertext, CMC, and collaboration support. OIS research was also allied with the technical side of information science, notably information retrieval and language processing.

The Formation of Association for Computing Machinery Special Interest Group on Computer–Human Interaction

Figure 1 identifies research fields that directly bear on HCI. Both HF and IS have distinct subgroups that focus on broad use of digital technologies. Relevant computer science research is concentrated in CHI, the subgroup primarily concerned with discretionary hands-on computer use. Other computer science influences—computer graphics, AI, office systems—have been described but are not included in Figure 1. The fourth field, information, began as support for specialists. It may come to exert the broadest influence of all.

Decreasing microcomputer prices encouraged discretionary hobbyists to use them. In 1980, as IBM prepared to launch the PC, a groundswell of attention on computer user behavior was building up. IBM, which like many hardware companies had not sold software separately, had decided to make software a product focus. Several cognitive psychologists joined an IBM group that included John Gould, who had been publishing human factors research since the late 1960s. They initiated empirical studies of programming and studies of software design and use. Other psychologists who in 1980 led recently formed HCI groups were Phil Barnard at the Medical Research Council Applied Psychology Unit in Cambridge, England; Tom Landauer at Bell Laboratories; Donald Norman at the University of California, San Diego; and John Whiteside at Digital Equipment Corp.

Xerox PARC and CMU collaborators continued research that led to an exceptionally influential project. The 1981 Star, with a carefully designed GUI, was not a commercial success

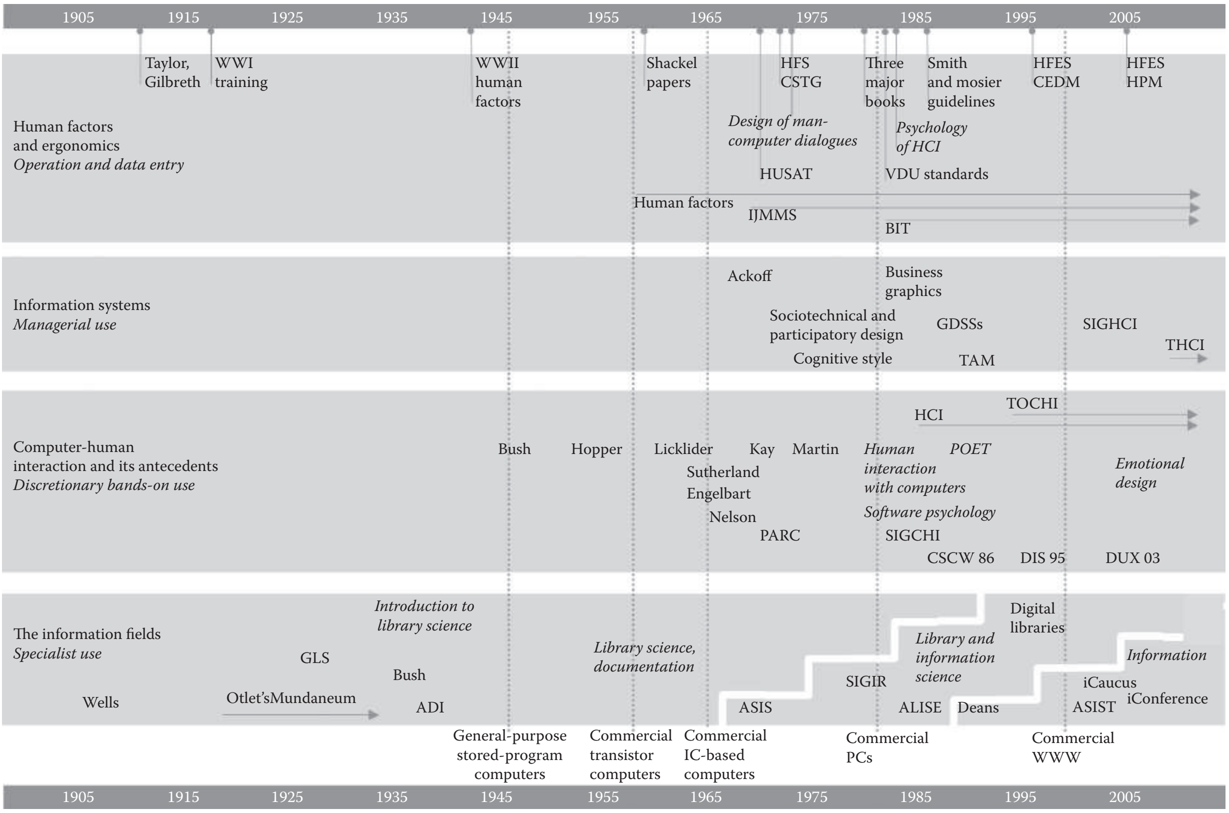


FIGURE 1 Four fields with major human–computer interaction research threads: Acronym expansions are provided in the text.

(nor were a flurry of GUIs that followed, including the Apple Lisa), but it influenced researchers and developers—and the design of the Macintosh.

Communications of the ACM created a "Human Aspects of Computing" department in 1980. The next year, Tom Moran edited a special issue of *Computing Surveys* on "The Psychology of the Computer User." Also in 1981, the ACM Special Interest Group on Social and Behavioral Science Computing (SIGSOC) extended its workshop to cover interactive software design and use. In 1982, a conference in Gaithersburg, Maryland, on "Human Factors in Computing Systems" was unexpectedly well attended. Shortly afterward, SIGSOC shifted its focus to Computer-Human Interaction and changed its name to SIGCHI (Borman 1996).

In 1983, the first CHI conference attracted more than 1000 people. Half of the 58 papers were from the aforementioned seven research laboratories. Cognitive psychologists in industry dominated the program, although the Human Factors Society cosponsored the conference and contributed the program chair Richard Pew; committee members Sid Smith, H. Rudy Ramsay, and Paul Green; and several presenters. Brian Shackel and HFS president Robert Williges gave tutorials on the first day. The International Conference on Human–Computer Interaction (INTERACT), first held in London in 1984 and chaired by Shackel, drew HF&E and CHI researchers.

The first profession to become discretionary hands-on users was computer programming, as paper coding sheets were discarded in favor of text editing at interactive terminals, PCs, and small minicomputers. Therefore, many early CHI papers, by Ruven Brooks, Bill Curtis, Thomas Green, Ben Shneiderman, and others, continued the psychology-of-programming research thread. Shneiderman formed the influential HCI Laboratory (HCIL) at Maryland in 1983. IBM researchers also contributed, as noted by John Thomas in a personal communication (October 2003): "One of the main themes of the early work was basically that we in IBM were afraid that the market for computing would be limited by the number of people who could program complex systems, so we wanted to find ways for 'nonprogrammers' to be able, essentially, to program."

Many experimental psychologists undertook studies of text editing, a tool initially used primarily by programmers. Thomas Green remarked at INTERACT'84 that "text editors are the white rats of HCI." As personal computing spread, studies of other discretionary use contexts were conducted. Studies of programming gradually disappeared from HCI conferences.

CHI focused on novice use. Initial experience is particularly important for discretionary users and for vendors developing software for them. Novice users are also a natural focus when studying new technologies and a critical focus when more people take up computing each year compared with the year before.

Routinized heavy use was still widespread. Databases were used by airlines, banks, government agencies, and other organizations. This hands-on activity was rarely discretionary. Managers oversaw development and analyzed data, leaving data entry and information retrieval to people hired for those jobs. To improve data management tasks was a human factors undertaking. CHI studies of database use were few—I count three over a decade, all focused on novice or casual use.

Fewer European companies produced mass-market software. European HCI research focused on in-house development and use, as reflected in the journal *Behaviour & Information Technology*, which was launched in 1982 by Tom Stewart and published by Taylor & Francis in London. In his perceptive essay cited in the section "Discretion in Computer Use," Bannon urged that more attention be paid to discretionary use, yet criticized CHI's heavy emphasis on initial experience, reflecting the European perspective. At Loughborough University, HUSAT focused on job design (the division of labor between people and systems) and collaborated with the Institute for Consumer Ergonomics, particularly on product safety. In 1984, Loughborough initiated an HCI graduate program drawing on human factors, industrial engineering, and computer science.

The work of the early visionaries was unfamiliar to many CHI researchers who were helping realize some of the early visions. The 633 references in the 58 papers presented at CHI'83 included many authored by cognitive scientists, but Bush, Sutherland, and Engelbart were not cited. A few years later, more computer scientists familiar with the early work joined CHI, notably those working on interactive computer graphics. The psychologists eventually discovered and identified with the pioneers, who shared their concern for discretionary use. This conceptual continuity bestowed legitimacy on a young enterprise that sought to establish itself academically and professionally.

Divergence of Computer–Human Interaction and Human Factors

> Hard science, in the form of engineering, drives out soft science, in the form of human factors.
>
> **Newell and Card (1985, p. 212)**

Between 1980 and 1985, Card, Moran, and Newell (1980a,b) introduced a "keystroke-level model for user performance time with interactive systems," followed by the cognitive model goals, operators, methods, and selection rules (GOMS) in their landmark 1983 book *The Psychology of Human–Computer Interaction*. This work was highly respected by the cognitive psychologists prevalent in CHI at the time. However, these models did not address discretionary, novice use. They focused on the repetitive expert use studied in human factors. In fact, GOMS was explicitly positioned to counter the latter field's stimulus–response bias: "Human-factors specialists, ergonomists, and human engineers will find that we have synthesized ideas from modern cognitive psychology and AI with the old methods of task analysis.… The user is not an operator. He does not operate the computer, he communicates with it" (Newell and Card 1985, p. viii.).

Newell and Card noted that HFs had a role in design, but continued: "Classical human factors … has all the earmarks of second-class status. (Our approach) avoids continuation of the classical human-factors role (by transforming) the psychology of the interface into a hard science" (p. 221).

In 2004, Card noted in an e-mail discussion: "Human Factors was the discipline we were trying to improve.… I personally changed the (CHI conference) call in 1986, so as to emphasize computer science and reduce the emphasis on cognitive science, because I was afraid that it would just become human factors again."

Ultimately, human performance modeling drew a modest but fervent CHI following. Key goals differed from those of other researchers and many practitioners. "The central idea behind the model is that the time for an expert to do a task on an interactive system is determined by the time it takes to do the keystrokes," wrote Card, Moran, and Newell (1980b, p. 397). Modeling was extended to a range of cognitive processes, but it was most useful in helping to design for nondiscretionary users such as telephone operators engaged in repetitive tasks (e.g., Gray et al. 1990). Its role in augmenting human intellect was unclear.

CHI and HFS moved apart, although "Human Factors in Computing Systems" remains the CHI conference subtitle. They were never highly integrated. Most of the cognitive psychologists had turned to HCI after earning their degrees and were unfamiliar with the human factors literature. The Human Factors Society did not again cosponsor CHI. Its researchers disappeared from the CHI program committee. Most CHI researchers who previously published in the human factors literature shifted to CHI, *Communications of the ACM*, and the journal *Human–Computer Interaction* launched in 1985 by Thomas Moran and published by Erlbaum, a publisher of psychology books and journals.

The shift was reflected at IBM T.J. Watson Research Center. John Gould and Clayton Lewis authored a CHI'83 paper that nicely framed the CHI focus on user-centered, iterative design based on prototyping. Cognitive scientists at Watson helped shape CHI, but Gould's principal focus remained human factors; he served as HFS president 4 years later. Reflecting the broader change, in 1984 the Human Factors Group at Watson began to dissolve and a User Interface Institute emerged.

CHI researchers, identifying with "hard" science or engineering, adopted the terms "cognitive engineering" and "usability engineering." In the first paper presented at CHI'83, "Design Principles for Human–Computer Interfaces," Donald Norman (1983) applied engineering techniques to discretionary use, creating "user satisfaction functions" based on technical parameters. These functions would not hold up long—people are fickle, yesterday's satisfying technology is not as gratifying today—but for years CHI emulated engineering, downplaying design, marketing, and other aspects of how humans interact with technology.

Workstations and Another Artificial Intelligence Summer

High-end workstations from Apollo, Sun, and Silicon Graphics appeared between 1981 and 1984. Graphics researchers no longer had to flock to heavily financed laboratories (notably MIT and Utah in the 1960s; MIT, New

York Institute of Technology, and PARC in the 1970s). Workstations were too expensive to reach a mass market, so graphics research that focused on photorealism and animation, which required the processing power of workstations, did not directly exert a broad influence on HCI.

The Xerox Star (formally named Office Workstation), Apple Lisa, and other commercial GUIs appeared, but when the first CHI conference was held in December 1983 none were commercial successes. They cost too much or ran on processors that were too weak to exploit graphics effectively.

In 1981, Symbolics and LMI introduced workstations optimized for the LISP programming language favored by most AI researchers. The timing was fortuitous. In October of that year, a conference on next-generation technology was held in the National Chamber of Commerce auditorium in Tokyo, Japan, and in 1982 the Japanese government established the Institute for New Generation Computer Technology (ICOT) and a 10-year fifth generation project focused on AI. AI researchers in Europe and the United States sounded the alarm. Donald Michie of Edinburgh saw a threat to Western computer technology, and in 1983 Ed Feigenbaum of Stanford and Pamela McCorduck wrote: "The Japanese are planning the miracle product.… They're going to give the world the next generation—the Fifth Generation—of computers, and those machines are going to be intelligent.… We stand, however, before a singularity, an event so unprecedented that predictions are almost silly.… Who can say how universal access to machine-intelligence—faster, deeper, better than human intelligence—will change science, economics, and warfare, and the whole intellectual and sociological development of mankind?" (pp. 8–9, 287).

Parallel distributed processing (often called neural networks) models also seized the attention of researchers and the media. Used for modeling phenomena including signal detection, motor control, and semantic processing, neural networks represented conceptual and technical advances over earlier AI work on perceptrons. Their rise was tied to the new generation of minicomputers and workstations, which had the power to support simulation experiments. Production systems, a computer-intensive AI modeling approach with a psychological foundation, developed at CMU, also gained the attention of researchers.

These developments triggered an AI gold rush. As with actual gold rushes, most of the money was made by those who outfitted and provisioned the prospectors, although generous government funding again flowed to the actual researchers. The European ESPRIT and UK Alvey programs invested over US$200 million per year starting in 1984 (Oakley 1990). In the United States, funding for the DARPA Strategic Computing AI program, begun in 1983, rose to almost US$400 million in 1988 (Norberg and O'Neill 1996). Investment in AI by 150 U.S. companies was estimated at about US$2 billion in 1985 (Kao 1998).

The unfulfilled promises of the past led to changes this time around. General problem solving was emphasized less, whereas domain-specific problem solving was emphasized more. Terms such as intelligent knowledge-based systems, knowledge engineering, expert systems, machine learning, language understanding, image understanding, neural networks, and robotics were often favored over AI.

In 1983, Raj Reddy of CMU and Victor Zue of MIT criticized the mid-1970s abandonment of speech-processing research, and soon funds again became plentiful for these research topics (Norberg and O'Neill 1996, p. 238). Johnson (1985) estimated that 800 corporate employees and 400 academics were working on natural language–processing research. Commercial natural language–understanding (NLU) interfaces to databases such as AI Corporation's Intellect and Microrim Clout appeared.

The optimism is illustrated by two meticulously researched Ovum reports on speech and language processing (Johnson 1985; Engelien and McBride 1991). In 1985, speech and language product "revenue" was US$75 million, comprising mostly income from grants and investor capital. That year, Ovum projected that sales would reach US$750 million by 1990 and US$2.75 billion by 1995. In 1991 sales were under US$90 million, but hope springs eternal and Ovum forecasts US$490 million for 1995 and US$3.6 billion for 2000.

About 20 U.S. corporations banded together, jointly funding the Microelectronics and Computer Technology Corporation (MCC). U.S. antitrust laws were relaxed to facilitate this cooperation. MCC embraced AI, reportedly becoming the leading customer for both Symbolics and LMI. MCC projects included two parallel NLU efforts; work on intelligent advising; and CYC (as in encyclopedic, and later spelled Cyc), Douglas Lenat's ambitious project to build a commonsense knowledge base that other programs could exploit. In 1984, Lenat predicted that by 1994 CYC would be intelligent enough to educate itself. Five years later, CYC was reported to be on schedule and about to "spark a vastly greater renaissance in [machine learning]" (Lenat 1989, p. 257).

Knowledge engineering involved human interaction. This could have brought AI closer to HCI, but AI researchers who were interested in representation and reasoning were frustrated by the difficulty of eliciting knowledge from experts. As many AI systems were aimed at nondiscretionary use, this created opportunities for HF&E, especially in Europe where funding directives dictated work that spanned technical and behavioral concerns. The journal *IJMMS* became a major outlet for both HF&E and AI researchers in the 1980s.

Interaction of AI and CHI was limited. CHI'83 and CHI'85 had a few sessions on speech and language, cognitive modeling, knowledge-based help, and knowledge elicitation. Not many AI researchers and developers worried about usability. They loved powerful tools such as EMACS and UNIX, forgetting the painful weeks required to learn the badly designed command languages. In general, AI technologies did not succeed in the marketplace. Before it disappeared, AI Corporation's primary customer for the database interface Intellect was the government, where discretionary use was not the norm.

1985–1995: GRAPHICAL USER INTERFACES SUCCEED

> "There will never be a mouse at the Ford Motor Company."
> **A high-level acquisition manager, 1985 (personal communication)**

When graphical user interfaces finally succeeded commercially, human–computer interaction was transformed. As with previous disruptive shifts—to stored programs and to interaction based on commands, full-screen forms, and full-screen menus—some people were affected before others. GUIs were particularly attractive to consumers, to new or casual users. Their success immediately transformed CHI, but only after Windows 3.0 succeeded in 1990 did GUIs influence the government agencies and business organizations that are the focus of HF&E and IS researchers. By 1990, the technology was better understood and thus less disruptive. The early 1990s also saw the maturation of local area networking and the Internet, producing a second transformation: computer-mediated communication and information sharing.

Computer–Human Interface Embraces Computer Science

Apple launched the Macintosh with a 1984 Super Bowl ad describing office work, but sales did not follow and by mid-1985 Apple was in trouble. Then Macs appeared with four times as much random access memory (RAM), which was sufficient to manage Aldus PageMaker, Adobe Postscript, the Apple LaserWriter, and Microsoft's Excel and Word for Macintosh as they were released. The more powerful Mac Plus arrived in January 1986. Rescued by hardware and software advances, the Mac succeeded where many commercial GUIs before it could not. It was popular with consumers and became the platform for desktop publishing.

Within CHI, GUIs were initially controversial. They had disadvantages: An extra level of interface code increased development complexity and created reliability challenges. They consumed processor cycles and distanced users from the underlying system that, many believed, experienced users must eventually master. Carroll and Mazur (1986) showed that GUIs confused and created problems for people familiar with existing interfaces. An influential 1986 essay on direct manipulation interfaces by Hutchins, Hollan, and Norman concluded that "it is too early to tell" how GUIs would fare. The GUIs could well prove useful for novices, they wrote, but "we would not be surprised if experts are *slower* with Direct Manipulation systems than with command language systems" (pp. 119-121, italics in the original). Given that most prior HCI research had focused on expert use, this insight seemed significant. However, first-time use proved critical in the rapidly expanding consumer market, and hardware and software improvements overcame some early limitations. GUIs were here to stay. CHI was soon transformed. Previously active research topics, including command naming, text editing, and the psychology of programming, were abandoned. More technical topics such as "user interface management systems" became significant.

Viewed from a higher plane, psychology gave way to computer science as the driving force in interaction design. Researchers had strived for a comprehensive, theoretical, psychological framework based on formal experiments (Newell and Card 1985; Carroll and Campbell 1986; Long 1989; Barnard 1991). Such a framework was conceivable for constrained command- and form-based interaction but could not be scaled to design spaces that included color; sound; animation; and an endless variety of icons, menu designs, and window arrangements. The new mission was to identify the most pressing problems and find satisfactory rather than optimal solutions. Rigorous experimentation, a skill of cognitive psychologists, gave way to quicker, less precise assessment methods championed by Jakob Nielsen (1989; Nielsen and Molich 1990).

Exploration of the dynamically evolving, relatively unconstrained design space required software engineering expertise. The late 1980s saw an influx of computer scientists to the CHI community. HCI entered the curricula of many computer science programs. CHI became a natural home to some computer scientists working on interactive graphics, software engineers interested in interaction, and AI researchers working on speech recognition, language understanding, and expert systems. In 1994, ACM launched the journal *Transactions on Computer–Human Interaction (TOCHI).* Early PCs and Macs were not easily networked, but as the use of local area networks spread, CHI's focus expanded to include collaboration support. This brought it into contact with efforts in MIS and OA research, discussed in the section on Collaboration Support below.

Human Factors and Ergonomics Maintains a Nondiscretionary Use Focus

Human factors and ergonomics research continued to respond to the needs of government agencies, the military, aviation industry, and telecommunications. Governments are the largest consumers of computing, for census, tax, social security, health and welfare, power plant operation, air traffic control, ground control for space missions, military logistics, text and voice processing for intelligence, and so on. The focus is on skilled use—users are assigned technology and trained if necessary. For routine data entry and other tasks, small efficiency gains in individual transactions can yield large benefits over time, justifying the effort to make improvements that might not be noticed by discretionary users. After SIGCHI formed, HFS undertook a study to see how CHI would affect membership in its Computer Systems Technical Group. An unexpectedly small effect was found (Richard Pew, personal communication; September 15, 2004). They had different goals.

Government agencies promoted the development of ergonomic standards to help in defining system requirements for competitive bidding while remaining at arms' length from potential developers, who of course better understood technical possibilities and helped with standards development.

Compliance with standards could then be specified in a contract. In 1986, Sid Smith and Jane Mosier published the last of a series of government-sponsored interface guidelines, with 944 design guidelines organized into sections titled Data Entry, Data Display, Data Transmission, Data Protection, Sequence Control, and User Guidance. The authors recognized that GUIs would expand the design space beyond the reach of this already cumbersome document that omitted icons, pull-down and pop-up menus, mice button assignments, sound, animation, and so on. Smith and Mosier foresaw that requirements definition must shift to specify predefined interface styles and design processes rather than features that would be built from scratch.

DARPA's heavily funded strategic computing AI program set out to develop an autonomous land vehicle, a pilot's associate, and a battle management system. All raised human factors research issues. These systems were to include interactive technologies such as speech recognition, language understanding, and heads-up displays. People might avoid these technologies when given a choice, but pilots guiding autonomous vehicles and officers under stressful conditions might have no better alternative. Speech and language technologies have other nondiscretionary potential, some of it civilian: for translators and intelligence analysts, when a phone system provides no alternative, when a disability limits keyboard use, or when hands are otherwise occupied.

Information Systems Extends Its Range

Although GUIs were not quickly adopted by organizations, spreadsheets and business graphics (charts and tables) were important to managers and thus the foci of IS research. Remus (1984) contrasted tabular and graphic presentations and Benbasat and Dexter (1985) added color as a factor, although color displays were rare in the 1980s. Many studies contrasted online and paper presentations, because most managers worked with printed reports. Although research into individual cognitive styles was abandoned in the early 1980s following a devastating critique on the topic (Huber 1983), the concept of cognitive fit between task and tool was introduced to explain apparently contradictory results in the adoption literature (Vessey and Galletta 1991).

A series of symposia on human factors in IS was initiated in 1986 by Jane Carey, leading to several books on the subject (e.g., Carey 1988). Topics included user interaction with information, design and development and, as corporate adoption of minicomputers and intranets matured, communication and collaboration, including studies of e-mail use.

The involvement of end users in the development process was actively discussed in IS, but rarely practiced outside of the sociotechnical design and the participatory design movements discussed below in the section "Participatory Design and Ethnography" (Friedman 1989). Hands-on managerial use was atypical in this period, but it was central to group decision support systems (GDSS) research. Central to GDSS was support for meetings, including brainstorming, idea organization, and online voting features. GDSS emerged from decision support systems, aimed at supporting individual executives or managers, and later evolved into group support systems. Computer-supported meeting facility research was conducted in the mid-1980s in several laboratories (e.g., Begeman et al. 1986; DeSanctis and Gallupe 1987; Dennis et al. 1988). Extensive research at the University of Arizona is summarized by Nunamaker et al. (1997). These systems were initially too expensive to be mass-market products; hence, the focus was on "decision makers," and research was conducted primarily in schools of management, not computer science departments or software companies. GDSS was a major IS contribution to computer-supported cooperative work (CSCW), discussed in the next section. In 1990, three companies began marketing GDSSs, including IBM and a University of Arizona spin-off, although without much success.

The Technology Acceptance Model (TAM) introduced by Davis (1989) led to considerable IS research. TAM and its offspring focus on perceived usefulness and perceived usability to improve "white-collar performance" that is "often obstructed by users' unwillingness to accept and use available systems" (p. 319). "An element of uncertainty exists in the minds of decision makers with respect to the successful adoption," wrote Bagozzi, Davis, and Warshaw (1992, p. 664). Although TAM is a managerial view of individual behavior, it was influenced by Davis's exposure to early CHI usability research.

TAM is probably the most cited HCI work in IS. The management view of hands-on computer use as nondiscretionary was giving way as use spread to white-collar workers who could refuse to play. TAM's emphasis on *perceived* utility and usability is a key distinction: Consumers choose technologies that they are convinced will be useful; CHI researchers assume utility and focuses on the *experience* of usability. TAM researchers focus on utility and note that perceptions of usability can influence acceptance. CHI addressed usability a decade before TAM, albeit actual usability rather than perceived usability. Perception was a secondary 'user satisfaction' measure to CHI researchers, who believed (not entirely correctly) that measurable reduction in time, errors, questions, and training would eventually translate into positive perceptions. The word "acceptance," that is, the "A" in TAM, is not in the CHI vocabulary. Discretionary users adopt, they do not accept.

The IS and CHI communities rarely mixed. When CHI was over a decade old, *Harvard Business Review*, a touchstone for IS researchers, published "Usability: The New Dimension of Product Design" (March 1994). The article did not mention CHI at all. It concluded that "user-centered design is still in its infancy" (p. 149).

Collaboration Support: Office Information Systems Gives Way to Computer-Supported Cooperative Work

In the late 1980s, three research communities addressed small-group communication and information sharing: (1) OA/OIS, described above in the section "Minicomputers and Office Automation." (2) IS researchers building systems to support organizational decision making could, as computing costs declined, address group decision making more generally. (3) The proliferation of local area networks enabled some CHI researchers to move from individual productivity software to the quest for "killer apps" that would support teams.

OA/OIS led the way, but it declined and was fast disappearing by 1995. The Minicomputers, the platform for most OIS research, did not survive competition from PCs and workstations. The concept of office or group proved to be problematic: Organizations and individuals are persistent entities with goals and needs, but small groups often have ambiguous membership and undergo shifts in character as members join or depart. People in an organization who need to communicate often fall under different budgets, complicating acquisition decisions unless a technology is made available organization-wide.

The rapid shift was reflected in terminology use. First, "automation" fell out of favor. In 1986, ACM SIGOA shifted to SIGOIS and the annual AFIPS OA conferences were discontinued. By 1991, the term "office" followed: *Transactions on Office Information Systems* became *Transactions on Information Systems; Office: Information and People* became *Information Technology and People*; and "Conference on Office Information Systems" became "Conference on Organizational Communication Systems" (COOCS, in 1997 becoming the GROUP Conference).

The AI summer, which contributed to the OA/OIS effort, ended when AI failed to meet expectations: Massive funding did not deliver a pilot's associate, an autonomous land vehicle, or a battle management system for the military. Nor were offices automated. CHI conference sessions on language processing had diminished prior to this AI winter, but sessions on modeling, adaptive interfaces, advising systems, and other uses of intelligence in interfaces increased through the 1980s before declining in the 1990s. Funding for AI became scarce, employment opportunities dried up, and conference participation dropped off.

A 1986 conference, building on a successful private 1984 workshop (Greif 1985), brought together researchers from diverse disciplines interested in issues of communication, information sharing, and coordination under the banner "Computer Supported Cooperative Work." Participants came primarily from IS, OIS, CHI, distributed AI, and anthropology. Four of 13 CSCW program committee members and many papers were from schools of management, with similar participation by the OIS community.

The field coalesced in 1988. The book *Computer-Supported Cooperative Work*, edited by Irene Greif, was published, and SIGCHI sponsored a biennial North American CSCW conference. A European series (ECSCW) was initiated in 1989. With heavy participation from technology companies, North American CSCW had a small-group focus on networked individuals working on PCs, workstations, or minicomputers. Groups were either within an organization or linked by ARPANET, BITNET, or other networks. European participation, primarily from academia and government agencies, focused on organizational use of technologies. It differed methodologically from most IS research in North America. Scandinavian influences, described in the next section, were felt in both CSCW and ECSCW.

Just as human factors researchers left CHI after a few years, most IS researchers who were involved with CSCW left in the early 1990s. One factor was a shift within IS from social psychology to organizational behavior in studying team behavior. The Hawaii International Conference on System Sciences (HICSS) was becoming a major IS prejournal publication venue for work with an organizational orientation. In contrast, the organizational focus conflicted with the CSCW interest in context-independent small-group support, which was the realm of social psychology and the goal of many technology companies. Some IS researchers participated in COOCS and GROUP. The split was not entirely amicable; the IS newsletter *Groupware Report* did not include CSCW on its list of relevant conferences.

The pace of technology change created challenges for CSCW. In 1985, supporting a small team was a technical challenge; 10 years later, the Web had arrived. Applications that provided awareness of the activity of distant collaborators was a celebrated achievement in the early 1990s; several years later, dark linings to the silver cloud arose in the form of privacy concerns and information overload. Phenomena, such as a "productivity paradox" in which IT investments were not returning benefits and health effects of Internet use by young people, were carefully identified only to vanish a few years later. Other changes brought European and North American CSCW into greater alignment. European organizations were starting to acquire commercial software products, a CSCW focus in North America, and North Americans were discovering that organizational context, an ECSCW focus, was often crucial in the design and deployment of products intending to support group activity. Organizational behaviorists and theorists were thriving in their home disciplines, but ethnographers studying technology use, marginalized in traditional anthropology departments, were welcomed into CSCW.

Despite the challenges of building on sands swept by successive waves of technology innovation, CSCW remains a strong research area that attracts a broad swath of HCI researchers. Content ranges from the highly technical to thick ethnographies of workplace activity, from studies of instant messaging dyads to scientific collaboratories involving hundreds of people dispersed in space and time. Chapter 24 by Gary and Judy Olson in this handbook covers the technical side of this topic in depth, with references to other CSCW resources.

Participatory Design and Ethnography

Prior to 1985-1995 some system developers explored methods to involve some of the future users in designing a system. Typically the users were nondiscretionary users of a system being developed by a large enterprise for its own use.

Sociotechnical design took a managerial perspective. Participatory or cooperative design, rooted in the Danish trade union movement, focused on empowering eventual users (Nygaard 1977).

Scandinavian approaches influenced human factors (e.g., Rasmussen 1986) and attracted wide notice with the publication of the proceedings of a conference held in Aarhus, Denmark, in 1985 (Bjerknes et al. 1987). Participatory design was a critique of IS approaches, yet the Scandinavians resonated with CHI researchers. Despite differences in culture, contexts of development (in-house system vs. commercial product), and contexts of use (nondiscretionary vs. discretionary), they shared the goal of empowering hands-on users. Most were also of the generation that grew up in the1960s, unlike the World War II generation that dominated HF&E and IS.

Ethnography was a different approach to obtaining deep insights into potential users. Lucy Suchman managed a Xerox PARC group that presented studies of workplace activity at CSCW. Suchman published an influential critique of artificial intelligence in 1987 and a widely read review of the Aarhus proceedings in 1988, and as program chair she brought many Scandinavians to the CSCW 1988 conference.

Library and Information Science: An Incomplete Transformation

Research universities have always supported prestigious professional schools, but the prestige of library schools declined with the rise of higher-paid IT and software engineering professions. Between 1978 and 1995, 15 American library schools were shut down (Cronin 1995, p. 45). Most of the survivors were rechristened Library and Information Science. The humanities orientation had given way, and librarianship was being changed by technology. New curricula and faculty with different skills were needed.

The changes did not go smoothly or as anticipated. Forced multidisciplinarity is never easy. Exclusion of technology studies may have been a reasonable reaction to the expense and limitations of new technologies. However, Moore's law lowered costs and removed many limitations with such speed that people and organizations had little time to prepare. Young information scientists were not interested in absorbing a century of work on indexing, classifying, and providing access to complex information repositories; their eyes were fixed on a future in which many past lessons would not apply. Those that still applied would likely have to be relearned. The conflicts are exposed in a landmark 1983 collection, *The Study of Information: Interdisciplinary Messages* (Machlup and Mansfield 1983). In the book, W. Boyd Rayward outlines the humanities-oriented perspective and the technological perspective and argues that there was convergence. His essay is followed by commentaries attacking him from both sides.

In a series of meetings beginning in 1988, new library and information school deans at the universities Pittsburgh, Syracuse, Drexel, and subsequently Rutgers discussed approaches to explaining and managing multidisciplinary schools. Despite this progressive effort, Cronin (1995) depicted LIS at loggerheads and in a "deep professional malaise." He suggested that librarianship be cut loose in favor of stronger ties to cognitive and computer sciences. Through the 1990s, schools at several universities dropped the word "library" and became schools of information (see Figure 2). More would follow.

1995–2010: THE INTERNET ERA ARRIVES

How did the spread of the Internet and the emergence of the Web affect HCI research threads? CHI researchers were Internet savvy. Although excited by the prospects, they took these changes in stride. Over time, CHI-related research, development, and use evolved. The Internet and the Web were not disruptive to HF&E either. The Web was initially a return to a form-driven interface style, and it was rarely a locus of routine work. However, the Web had a seismic impact on IS and on information science, so this section begins with these disciplines.

The Formation of Association for Information Systems Special Interest Group in Human–Computer Interaction

The use of computers in organizations has changed. Organizations are no longer focused on maximizing computer use—almost everywhere, screen savers have become the main consumer of processor cycles. Advent of the Internet created more porous organizational boundaries. Employees in many organizations could download software such as instant-messaging clients, music players, and weblog tools inside organizational firewalls despite IT concerns about productivity and security. These are not the high-overhead applications of the past. Increasingly, software can be used from a web browser without requiring a download. Experience with all of this at home leaves employees impatient with poor software at work. In addition, many managers who had been hands-off users became late adopters in late 1990s or were replaced by younger managers. Today, managers and executives are hands-on early adopters of many technologies.

Significant as these changes are, the Web had a more dramatic effect on organizational information systems. Corporate IT groups had been focused solely on internal operations. They lived inside firewalls. Their customers were other employees. Suddenly, organizations were scrambling to create Web interfaces to external vendors and customers. Discretionary users! The Internet bubble burst, revealing that IT professionals, IS experts, and everyone else had limited understanding of Web phenomena. Nevertheless, online marketing, services, and business-to-business systems continued to grow. For many, the Web had become an essential business tool. In handling external customers, IT professionals and IS researchers were in much the same place that CHI was 20 years earlier, whether they realized it or (most often) not.

In 2001, the Association for Information Systems (AIS) established a Special Interest Group in Human–Computer

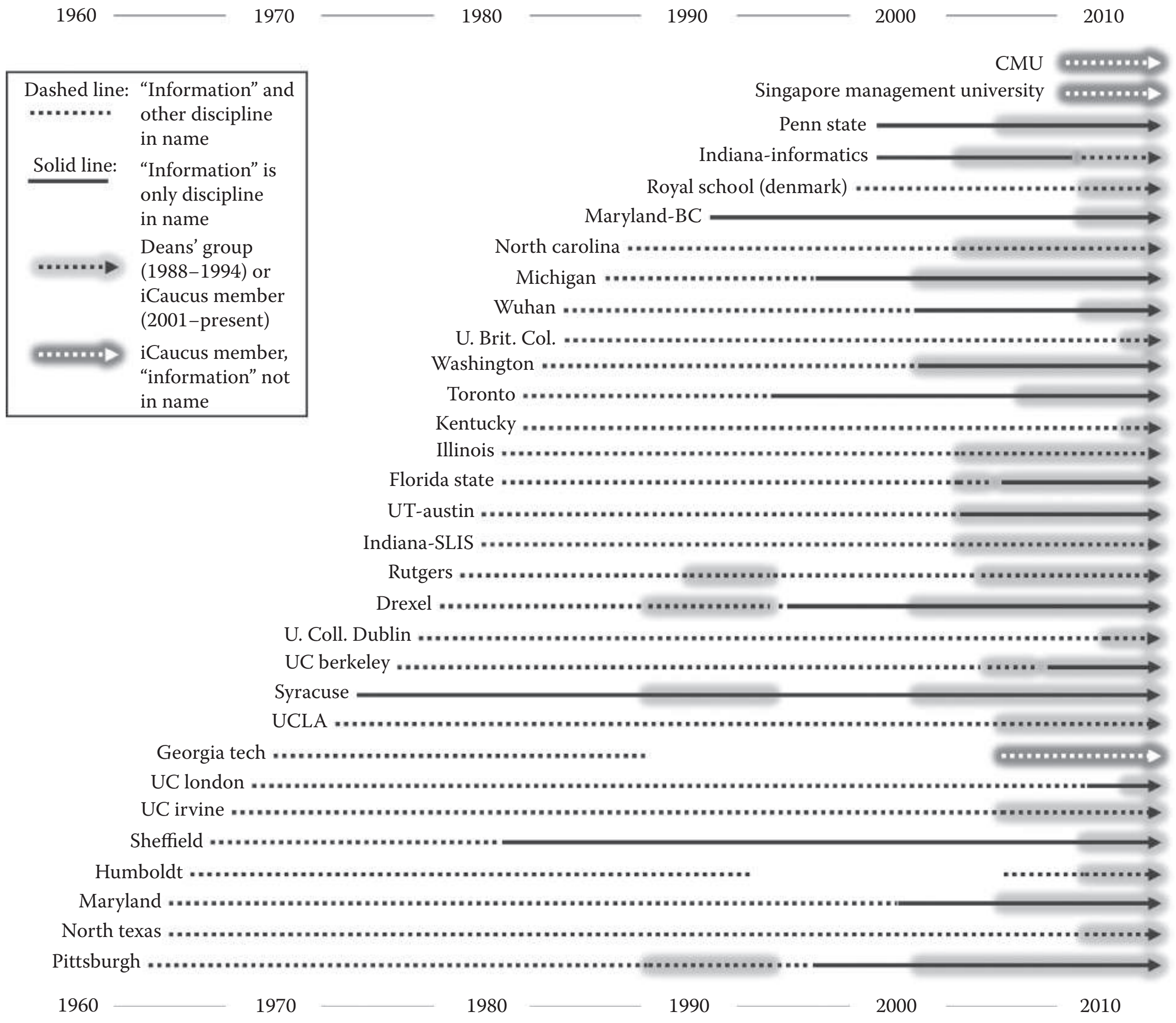


FIGURE 2 The iSchools and when "information" came into the names of the member Schools (Faculties, Colleges, etc.).

Interaction (SIGHCI). The founders defined HCI by citing 12 CHI research papers (Zhang, Nah, and Preece 2004, p. 148). Bridging the CHI and the information science communities was declared a priority. The charter of SIGHCI includes a broad range of organizational issues, but the publications emphasize interface design for e-commerce, online shopping, online behavior "especially in the Internet era" (Zhang 2004, p. 1), and effects of Web-based interfaces on attitudes and perceptions. Eight of the first 10 papers in SIGHCI-sponsored journal issues covered Internet and Web behavior.

In 2009, the journal *AIS Transactions on Human–Computer Interaction* was launched. The shift from an organizational focus to the Web and broader end-user computing is documented in Zhang et al.'s analysis (2009) of the IS literature from 1990 to 2008. This survey omits CHI from a list of the fields related to AIS SIGHCI. The bridging effort had foundered, as had three previous efforts to bridge to CHI: from Human Factors, Office Information Systems, and the Information Systems presence within CSCW.

Digital Libraries and the Evolution of Library Information Science

By 1995, an information wave had swept through universities (Figure 2). Digital technology was in the LIS curriculum. Familiarity with technology use was a prerequisite for librarianship. However, innovative research had not kept pace with professional training (Cronin 1995).

The Internet grew exponentially, but in 1995 it was still a niche activity found mainly on campuses. In the mid-1990s, Gopher, a convenient system for downloading files over the Internet, attracted attention as a possible springboard for indexing distributed materials. Wells's (1938) concept of "world brain" seemed to be within reach. Then the Web hit, transforming information acquisition, management, and access at an ever-increasing pace. Between 1994 and 1999, two NSF/DARPA/NASA/National Library of Medicine/Library of Congress/National Endowment for the Humanities/FBI initiatives awarded close to US$200 million for digital libraries research and development. This and other investments galvanized the research community. In 2000, the American Society for Information Science appended "and Technology" to its name to become ASIST.

By 2000, 10 schools (or equivalent units) had information as the sole discipline in their name. In 2001 a series of deans meetings began, which were modeled on those of the late 1980s. The original members, Syracuse, Pittsburgh, and Drexel, were joined by Michigan; Berkeley, California; and the University of Washington. All are now information schools. In 2005, the first annual "iConference" drew participants from 19 universities with information programs. As of 2011, the "iCaucus" had 27 dues-paying members. Some are transformed library schools, some have closer ties with other disciplines, and some have formed recently as schools of information. Collectively, their faculty includes HCI researchers trained in each of the four disciplines highlighted in this introduction.

Expansion is not without growing pains. Conflicts arise among academic subcultures. The iConference competes with more established conferences in each field. Figure 2 suggests that a shift to a field called information is well underway, but many faculty still consider themselves "a researcher in {X} who is located in an information school," where X could be library science, HCI, CSCW, IS, communication, education, computer science, or another discipline. We do not know how it will evolve, but we can say with confidence that information has become, and will remain, a significant player in HCI.

Human Factors and Ergonomics Embraces Cognitive Approaches

In 1996, the HFES formed a new technical group, Cognitive Engineering and Decision Making. It quickly became the largest technical group. A decade earlier this would have been unthinkable: Some leading human factors researchers disliked cognitive approaches. The CHI community first used the term cognitive engineering in this sense (Norman 1982, 1986). As this development suggests, CSTG declined in size and prominence as the HCI community dispersed. Most HF&E technical groups, from groups on telecommunications to those on medical systems, address digital technology and thereby HCI-related research.

Equally astonishing, in 2005 Human Performance Modeling was a new and thriving HFES technical group, initiated by Wayne Gray and Dick Pew, who had been active in CHI in the 1980s. Card, Moran, and Newell (1983) had introduced human performance modeling to reform the discipline of Human Factors from without. Some work continued within CHI that was focused on expert performance (e.g., a special issue of *Human–Computer Interaction*, vol. 12, number 4, 1997), but today the reform effort has moved within HF&E and remains focused largely on nondiscretionary use.

Government funding of HCI was largely shaped by the focus of HF&E. The Interactive Systems Program of the U.S. NSF—subsequently renamed HCI—was described thus: "The Interactive Systems Program considers scientific and engineering research oriented toward the enhancement of human–computer communications and interactions in all modalities. These modalities include speech/language, sound, images and, in general, any single or multiple, sequential, or concurrent, human–computer input, output, or action" (National Science Foundation 1993).

One NSF program manager identified his proudest accomplishment to be doubling the already ample funding for natural language understanding research. Even after NSF established a separate Human Language and Communication Program in 2003, speech and language research was heavily supported by both the HCI and accessibility programs, with lighter support from AI and other programs. Subsequent NSF HCI program managers emphasized "direct brain interfaces" or "brain–computer interaction" based on brain waves and implants. A review committee noted that a random sample of NSF HCI grants included none by prominent CHI researchers (National Science Foundation 2003). NSF program managers rarely attended CHI conferences, which have little coverage of speech, language, or direct brain interaction. These technologies may prove useful, but they have so far made few inroads into discretionary use situations in homes and offices.

Computer–Human Interaction Evolves, and Embraces Design

The steady flow of new hardware, software features, applications, and systems ensures that people are always encountering and adopting digital technologies for the first time. This is important for technology producers and it generates new research issues. CHI has tracked this, generally focusing on an innovation when it first starts to attract a wide audience.

As an application matures, its use often becomes routine. Technologies such as e-mail and word processing, no longer discretionary for most of us, get less attention from CHI researchers whose gaze is directed toward the discretionary use of the moment, including Web design, ubiquitous and mobile computing, social computing, and use of Wikipedia. New issues include information overload, privacy, and effects of multitasking, and encourage the emergence of new methods, such as ethnography and data mining. At a higher level, continuity is found in CHI: exploration of input devices, communication channels, information visualization techniques, and design methods. Proposals to build HCI theory on these shifting sands (Barnard et al. 2000; Carroll 2003) remain largely aspirational.

Expanding participation in the Internet as its reliability and bandwidth increased steadily through the mid-1990s brought real-time and quasi-real-time communication technologies such as e-mail into greater focus. The Web temporarily slowed this by shifting attention to indirect interaction with static sites, but with the advent of Web 2.0 and greater support for animation and video the pace quickened. The Web was like a new continent. Explorers posted flags here and there. Then came attempts at settlement, with the virtual worlds research and development that blossomed in the late

1990s. Few of the early pioneers survived; there was little to do in virtual worlds other than chat and play games. But slowly some people shifted major portions of their work and play online, relying on online information sources, digital photo management, social software, digital documents, online shopping, multiplayer games, and so on. This evolution is reflected in CHI research.

The content of CSCW in North America has shifted in response to the extraordinary growth of social networking sites, Wikipedia, and other Web phenomena, which are of intense interest to students and academic researchers and the software companies who hire or consult with many of them. These technologies are not yet of great interest to the organizations and government agencies that are the customer for European CSCW research, and the move toward shared interests has been reversed. Europeans have moved more rapidly into basic research in vertical domains. The division resembles that of 20 years ago, based on a new generation of technology. In several years the two research threads may again converge, perhaps under different names: "computer supported cooperative work" is outdated. Many digital devices are not considered computers, they play central rather than support roles, activities around them can be competitive or conflictual, and they may be used more for recreation than work.

The Web curtailed research into one thread of AI research: powerful, self-contained personal productivity tools. Considerable effort is required to embed knowledge in application software, but when access to external information sources was limited, it was worth trying. With today's easy access to information and knowledgeable people online, static, self-contained knowledge representation is less useful. In contrast, adaptive systems that merge and filter local and Internet-based information have a role to play. Steady progress in machine learning is enhancing productivity tools, although implausible AI forecasts have not disappeared.

To the psychologists and computer scientists who formed the CHI community, interface design was a matter of science and engineering. They focused on performance and assumed that people eventually choose efficient alternatives. Because human discretion involves aesthetic preferences and invites marketing and nonrational persuasion, this view was not sustained when computing costs came down. This engineering orientation gripped CHI longer than SIGGRAPH, where aesthetic appeal was a major driver. CHI researchers eventually came around, labeling the study of enjoyment "funology" (Blythe et al. 2003) lest someone think that they were having too good a time.

Some visual designers participated in graphical interface research early on. Aaron Marcus began working full time on computer graphics in the late 1960s. William Bowman's book *Graphic Communication* (1968) was a strong influence on the development of Xerox Star, for which the designer Norm Cox's icons were chosen (Bewley et al. 1983). However, graphic design was considered a secondary activity (Evenson 2005). In 1995, building on workshops at previous conferences, SIGCHI initiated "Designing Interactive Systems" (DIS), a biennial conference that draws more systems designers than visual designers. In 2003, SIGCHI, SIGGRAPH, and the American Institute of Graphic Arts (AIGA) initiated the "Designing for User Experience" (DUX) conference series that fully embraced visual and commercial design. This effort lasted only through 2007, but the significance of design was established. Design is not typically assessed in research papers. The changing sensibility is reflected in *ACM Interactions*, a magazine launched by CHI in 1994, which has steadily increased the focus on design in both its content and its appearance.

Design's first cousin, marketing, has been poorly regarded by the CHI community (Marcus 2004). Website design forced the issue. Site owners wish to keep users interested in a site, whereas users may prefer to escape quickly. Consider supermarkets, which position items that most shoppers want far apart, forcing people to traverse aisles where other products beckon. CHI professionals who align themselves with end users face a stakeholder conflict when designing for a site owner. This was not true in the past: Designers of individual productivity tools had little conflict of interest with prospective customers. Marketing is concerned with identifying and satisfying user needs, as well as shaping them. It will likely find a place in CHI, perhaps labeled "brandology."

Finally, CHI has gradually become more open to work that takes a social or political stance. Accessibility was first addressed in the context of physical constraints. Socioeconomic factors were included in Universal Usability conferences in 2000 and 2003. Sustainability and fitness emerged as topics. This may reflect a distancing from a sense that engineering should strive for value neutrality, a bid for relevance by an increasingly academic group or aging CHI baby boomers who are considering their legacies.

The evolution of CHI is reflected in the influential contributions of Donald Norman. A cognitive scientist who introduced the term cognitive engineering, he presented the first CHI'83 paper. It defined "user satisfaction functions" based on speed of use, ease of learning, required knowledge, and errors. His influential book *Psychology of Everyday Things* (1988) focused on pragmatic usability. Its 1990 reissue as *Design of Everyday Things* reflected a field refocusing on invention. Fourteen years later he published *Emotional Design: Why We Love (or Hate) Everyday Things*, stressing the role of aesthetics in our response to objects.

LOOKING BACK: CULTURES AND BRIDGES

Despite overlapping interests, in a dynamic environment with shifting alliances, the major threads of HCI research—HF&E, IS, LIS, and CHI—have not merged. They have interacted with each other only sporadically, although not for a lack of bridge-building efforts. The Human Factors Society co-organized the first CHI conference. CSCW sought to link CHI and IS. Mergers of OIS with CHI and later CSCW were considered. AIS SIGHCI tried to engage with CHI. Researchers recently hired into Information Schools remain active in the other fields.

Even within computer science, bridging is difficult. Researchers interested in interaction left SIGGRAPH to join the CHI community rather than form a bridge. A second opportunity arose 20 years later, when standard platforms powerful enough to support photorealism loomed, but the DUX conference series managed only three meetings. For AI, SIGART and SIGCHI cosponsor the Intelligent User Interface series, but participation has remained outside mainstream HCI. What are the obstacles to more extensive interaction across fields?

Discretion as a Major Differentiator

HF&E and IS arose before discretionary hands-on use was common. The information field only slowly distanced itself from supporting specialists. CHI occupied a new niche: discretionary use by nonexperts. HF&E and especially IS researchers considered organizational factors; CHI with few exceptions avoided domain-dependent work. As a consequence, HF&E and IS researchers shared journals. For example, Benbasat and Dexter (1985) published their work in *Management Science* and cited five *Human Factors* articles. Apart from LIS, they quickly focused on broad populations. IS countered its organizational focus by insisting that work be framed by theory, which set it apart from CHI in particular.

The appropriateness of a research method is tied to the motivation of the researchers. HF&E and CHI were shaped by psychologists trained in experimental testing of hypotheses about behavior, and hypothesis-driven experimentation was also embraced by IS. Experimental subjects agree to follow instructions for an extrinsic reward. This is a reasonable model for nondiscretionary use, but not for discretionary use. CHI researchers relabeled subjects as "participants," which sounds volitional, and found that formal experimental studies were usually inappropriate: There were too many variables to test formally and feedback from a few participants was often enough. Laboratory studies of initial or casual discretionary use usually require confirmation in real-world settings anyway, more so than studies of expert or trained behavior, because of the artificial motivation of the laboratory study participant.

The same goals apply—fewer errors, faster performance, quicker learning, greater memorability, and being enjoyable—but the emphasis differs. For power plant operation, error reduction is critical, performance enhancement is good, and other goals are less important. For telephone order entry takers performance is critical, and testing an interface that could shave a few seconds from a repetitive operation requires a formal experiment. In contrast, consumers often respond to visceral appeal and initial experience. In assessing designs for mass markets, avoiding obvious problems can be more important than striving for an optimal solution. Less rigorous discount usability or cognitive walk-through methods (Nielsen 1989; Lewis et al. 1990) can be enough. Relatively time-consuming qualitative approaches, such as contextual design or persona use (Beyer and Holtzblatt 1998; Pruitt and Adlin 2006), can provide a deeper understanding when context is critical or new circumstances arise.

CHI largely abandoned its roots in scientific theory and engineering, which does not impress researchers from HF&E or theory-oriented IS. The controversial psychological method of verbal reports, developed by Newell and Simon (1972) and foreshadowed by gestalt psychology, was applied to design by Clayton Lewis as "thinking aloud" (Lewis and Mack 1982; Lewis 1983). Perhaps the most widely used CHI method, it led some researchers in other fields to characterize CHI people as wanting to talk about their experiences instead of doing research.

Academic, Linguistic, and Generational Cultures

The academic culture of the sciences is that conferences are venues for work in progress and journals are repositories for polished work. The disciplines of HF&E, IS, Documentation, and Library Science adhere to this practice. In contrast, for U.S. computer science disciplines, conference proceedings are now the final destination of most work. Outside the United States, computer science retains a journal focus, which suggests that a key factor was the decision of ACM to archive conference proceedings (Grudin 2010). Information science draws on researchers from both camps, journals as archival and conferences as archival. Of course, a difference in preferred channel impedes communication. Researchers in journal cultures chafe at CHI's insistence on polish and its high conference rejection rates; CHI researchers are dismayed by the lack of polish at other conferences and are less inclined to read journals.

CHI conferences accept 20%–25% of submissions. With a few exceptions, HF&E and IS conferences accept about 50% or more. In contrast, CHI journals receive fewer submissions and have higher acceptance rates. Many CHI researchers report that journals are not relevant. By my estimate, at most 15% of the work in CHI-sponsored conferences reaches journal publication. In contrast, an IS track organizer for HICSS estimated that 80% of research there progressed to a journal (Jay Nunamaker, opening remarks at HICSS-38, January 2004).

A linguistic divide also set CHI apart. HF&E and IS use the term "operator" and a "user" could be a manager who read printed reports. For CHI, "operator" was demeaning and a "user" was always a hands-on user. In HF&E and IS streams, "task analysis" refers to an organizational decomposition of work, perhaps considering external factors; in CHI, "task analysis" is a cognitive decomposition, such as breaking a text editing move operation into select, cut, select, and paste. In IS "implementation" means organizational deployment, whereas in CHI it is a synonym for development. The terms "system," "application, " and "evaluation" also have different connotations or denotations in the different fields. Significant misunderstandings resulted from failures to appreciate these differences.

Different perspectives and priorities were also reflected in attitudes toward standards. Many HF&E researchers contributed to standards development, believing that standards contribute to efficiency and innovation. A view widespread in the CHI community was that standards inhibit innovation. Both views have elements of truth, and the positions partly converged as Internet and Web standards were tackled. However, the attitudes reflected the different demands of government contracting and commercial software development. Specifying adherence to standards is a useful tool for those preparing requests for proposals, whereas compliance with standards can make it more difficult for a product to differentiate itself.

The generational divide was also a factor. Many CHI researchers who grew up in the 1960s and 1970s did not appreciate the prior generation's orientation toward military, government, and business systems. They were also put off by the lack of gender neutrality in the HF&E and IS "man–machine interaction" literature, which one still occasionally encounters. Only in 1994 did *IJMMS* become *International Journal of Human–Computer Studies*. Such differences affected the enthusiasm for building bridges and exploring literatures.

Competition for resources was another factor. Computers of modest capability were extremely expensive for much of the time span we have considered. CHI was initially largely driven by the healthy tech industry, whereas research in the other fields was more dependent on government funding that waxed and waned. When funding waxed, demand for researchers outstripped supply. HCI prospered during AI winters, starting with Sutherland's use of the TX-2 when AI suffered its first setback and recurring with the emergence of major HCI laboratories during the severe AI winter of the late 1970s. Library schools laboring to create information science programs had to compete with computer science departments that awarded faculty positions to graduates of master's programs when the supply was low.

Greater interdisciplinarity is intellectually seductive. Could we not learn by looking over fences? But a better metaphor might be the big bang. Digital technology is an explosion, streaming matter and energy in every direction, forming worlds that at some later date might discover one another and find ways to communicate, and then again, might not.

LOOKING FORWARD: TRAJECTORIES

The future of HCI will be dynamic and full of surprises. The supralinear growth of hardware capability confounds efforts at prediction: We rarely experience exponential change and do not reason well about it. In the United States, NSF is tasked with envisioning the future and providing resources for taking us there, yet two major recent HCI initiatives, "Science of Design" and "CreativIT" (focused on creativity), wound down quickly. Nevertheless, extrapolations from observations about the past and present suggest possible developments, providing a prism through which to view other chapters in this handbook and perhaps some guidance in planning a career.

Discretion: Now You See It, Now You Don't

We exercise prerogative when we use digital technology—sometimes. More often when at home, less often at work. Sometimes we have no choice, as when confronted by a telephone answering system. Those who are young and healthy have more choices than those constrained by injury or aging.

Many technologies follow the maturation path shown in Figure 3. Software that was discretionary yesterday is indispensable today. Collaboration forces us to adopt shared conventions. Consider a hypothetical team that has worked together for 20 years. In 1990, members exchanged printed documents. One person still used a typewriter, whereas others used different word processors. One emphasized words by underlining, another by italicizing, and a third by bolding. In 2000, the group decided to exchange digital documents. They had to adopt the same word processor. Choice was curtailed; it was only exercised collectively. Today this team is happy sharing documents in PDF format, so they can again use different word processors. Perhaps tomorrow software will let them personalize their view of a single underlying document, so one person can again use and see in italics what another sees as bold or underlined.

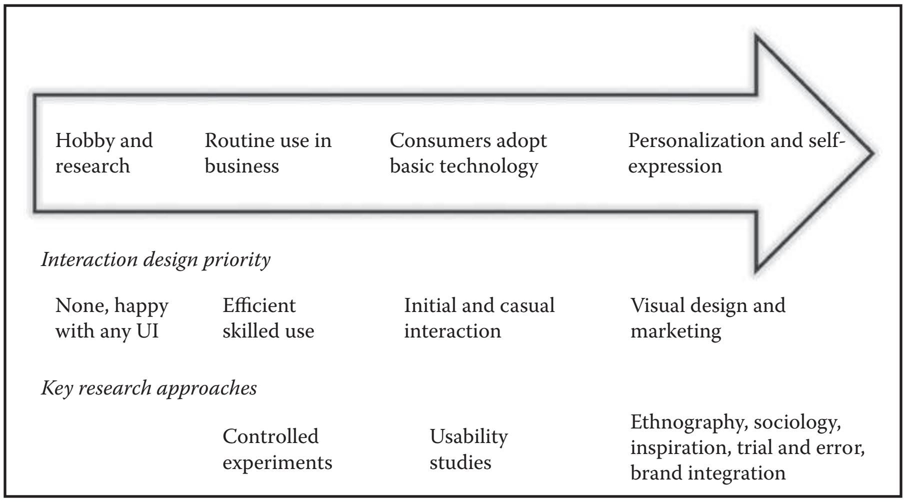

Hobby and research
Routine use in business
Consumers adopt basic technology
Personalization and self-expression
Interaction design priority
None, happy with any UI
Efficient skilled use
Initial and casual interaction
Visual design and marketing
Key research approaches
Controlled experiments
Usability studies
Ethnography, sociology, inspiration, trial and error, brand integration

FIGURE 3 From invention to maturity.

Shackel (1997, p. 981) noted this progression under the heading “From Systems Design to Interface Usability and Back Again.” Early designers focused at the system level; operators had to cope. When the PC merged the roles of operator, output user, and program provider, the focus shifted to the human interface and choice. Then individual users again became components in fully networked organizational systems. Discretion can evaporate when a technology becomes mission-critical, as word processing and e-mail did in the 1990s.

The converse also occurs. Discretion increases when employees can download free software, bring smartphones to work, and demand capabilities they enjoy at home. Managers are less likely to mandate the use of a technology that they use and find burdensome. For example, language-processing systems appealed to military officers, until they themselves became hands-on users:

> Our military users … generally flatly refuse to use any system that requires speech recognition.… Over and over and over again, we were told “If we have to use speech, we will not take it. I don’t even want to waste my time talking to you if it requires speech.…” I have seen generals come out of using, trying to use one of the speech-enabled systems looking really whipped. One really sad puppy, he said “OK, what’s your system like, do I have to use speech?” He looked at me plaintively. And when I said “No,” his face lit up, and he got so happy (Forbus 2003; see also Forbus, Usher, and Chapman [2003]).

In domains where specialized applications become essential and where security concerns curtail openness, discretion can recede. But Moore’s law (broadly construed), competition, and the ease of sharing bits should guarantee a steady flow of experimental technologies with unanticipated and thus initially discretionary uses.

Ubiquitous Computing: Invisible Human–Computer Interaction?

Norman (1988, p. 185) wrote of “the invisible computer of the future.” Like motors, he speculated, computers would be present everywhere and visible nowhere. We interact with clocks, refrigerators, and cars. Each has a motor, but who studies human–motor interaction? Marc Weiser subsequently introduced a similar concept, “ubiquitous computing.” A decade later, at the height of the Y2K crisis and the Internet bubble, computers were more visible than ever. But after a quarter century, while we may always want a large display or two, would anyone call a smartphone or a book reader a computer? The visions of Norman and Weiser may be materializing.

With digital technology embedded everywhere, concern with interaction is everywhere. HCI may become invisible through omnipresence. As interaction with digital technology becomes part of everyone’s research, the three longstanding HCI fields are losing participation.

Human Factors and Ergonomics

David Meister, author of *The History of Human Factors and Ergonomics* (1999), stresses the continuity of HF&E in the face of technology change:

> Outside of a few significant events, like the organization of HFS in 1957 or the publication of Proceedings of the annual meetings in 1972, there are no seminal occurrences … no sharp discontinuities that are memorable. A scientific discipline like HF has only an intellectual history; one would hope to find major paradigm changes in orientation toward our human performance phenomena, but there is none, largely because the emergence of HF did not involve major changes from pre-World War II applied psychology. In an intellectual history, one has to look for major changes in thinking, and I have not been able to discover any in HF (e-mail, September 7, 2004).

Membership in the Computer Systems Technical Group has declined. Technology is heavily stressed in technical groups such as Cognitive Engineering and Decision Making, Communication, Human Performance Modeling, Internet, System Development, and Virtual Environment. Nor do Aging, Medical Systems, or other technical groups avoid “invisible computers.”

Information Systems

While IS was thriving during the Y2K crisis and the Internet bubble, other management disciplines—finance, marketing, operations research, and organizational behavior—became more technically savvy. When the bubble burst and enrollments declined, the IS niche became less well defined. The research issues remain significant, but this cuts two ways. As IT organizations standardize on products and outsource IT functions, business-to-business and web portals for customers get more attention. These give rise to finance and marketing considerations, so HCI functions could be assumed by other management disciplines.

Computer–Human Interaction

This nomadic group started in psychology and then won a seat at the computer science table, which was bestowed grudgingly. Several senior CHI people moved to information schools. Lacking a well-defined academic niche, CHI ties its identity to the SIGCHI organization and the CHI conference. Membership in SIGCHI peaked in 1992 and conference attendance peaked in 2001. As new technologies become widely used, specialized conferences appear, often started by younger researchers. World Wide Web conferences included papers on HCI issues from the outset. HCI is an “invisible” presence in conferences on agents, design, and on computing that is ubiquitous, pervasive, accessible, social, and sustainable. High rejection rates for conference submissions and a new generational divide could accelerate the dispersion of research.

CHI attendance has become more exclusively academic, despite industry’s need for basic research in specific areas.

Apart from education and health, which have broad appeal, and software design and development, CHI remains largely focused on general phenomena and resistant to domain-specific work. This creates additional opportunities for regional and specialized conferences.

Information

Early in the computer era, there were no networks and memory was fantastically expensive. Computers were for computation, not information processing. Today, the situation is reversed: Memory and bandwidth are so plentiful that most computation is in the service of processing and distributing information. And the shift to an emphasis on information, with computation present but less visible, could well accelerate.

Cronin (1995) proposed that information access, in terms of intellectual, physical, social, economic, and spatial/temporal factors, is the focus of the information field. Information is acquired from sensors and human input; it flows over networks including the Web, and is aggregated, organized, and transformed. The routing and management of information within enterprises, as well as the consequences of ever more permeable organizational boundaries, is evolving. Approaches to personal information management are also rapidly changing. It was once centered on shoeboxes of photographs and boxes of old papers. Now most of us face significant online information management decisions, choosing what to keep locally, what to maintain in the cloud, and how to organize it to ensure its future accessibility. The CHI field has over a decade of work on information design and visualization (see Chapter 23 by Stuart Card).

In speculating about the future, Cronin (1995, p. 56) quotes Wersig (1992) who argued that concepts around information might function "like magnets or attractors, sucking the focus-oriented materials out of the disciplines and restructuring them within the information scientific framework." Could this happen? Information schools have hired senior and junior people from many relevant areas. Andrew Dillon, dean of the University of Texas, School of Information, worked at Loughborough with Brian Shackel and Ken Eason. Syracuse, the first extant school of information (since 1974), has faculty with IS training and orientation. CHI faculty have migrated to information schools and departments of several leading universities.

Communication Studies is a discipline to watch. Rooted in humanities and social sciences, it is gradually assuming a quantitative focus. Centered on studies of television and other mass media, the field blossomed in the 1980s and 1990s. Only in the last several years has computer-mediated communication reached the scale of significance of other mass media. HCI is in a position to draw on past work in communication, as communication focuses more on digital media.

The rise of specialized programs—biomedical informatics, social informatics, community informatics, and information and communication technology for development (ICT4D)—works against the consolidation of information studies. Information, like HCI, could become invisible through ubiquity. The annual Information Conference is a barometer. In 2005 and 2006, there was active discussion and disagreement about directions. Should new journals and research conferences be pursued, or should the field stick with the established venues in the various contributing disciplines? In the years since, faculty from different fields worked out pidgin languages with which to communicate with each other. Assistant professors were hired and graduate students enlisted, whose initial jobs and primary identities are with information. Will they creolize the pidgin language?

One can get a sense that the generals may still be arguing over directions, but the troops are starting to march. It is not clear where they will go. The generals, although busy with local campaigns, are reluctant to turn over command. The annual iConference vies with the less international but more established ASIST conference. However this evolves, in the long term, information is likely to be the major player in HCI. Design and information are active foci of HCI today, but the attention to design is compensation for past neglect. Information is being reinvented.

CONCLUSION: THE NEXT GENERATION

Looking back, cyclic patterns and cumulative influences are visible. New waves of hardware enabled different ways to support the same activities. E-mail arrived as an informal communication medium, was embraced by students, regarded with suspicion by organizations, and eventually became more formal and used everywhere. Then texting and instant messaging came along as an informal medium, were embraced by students, regarded with suspicion by organizations, and eventually became used everywhere. Social networking came along. …

Mindful of Edgar Fiedler's admonition that "he who lives by the crystal ball soon learns to eat ground glass," consider this: In the mid-1980s, the mainframe market lost the spotlight. Organizations were buying hundreds of PCs, but these were weak devices with little memory, hard to network. They did not need more mainframes, but what about a massive, parallel supercomputer? Government and industry invested vast sums in high-performance computing only to discover that it was hard to decompose most computational problems into parallel processes whose output could be reassembled. As these expensive and largely ineffective efforts proceeded, PCs slowly got stronger, added some memory, got networked together, and, without vast expenditures and almost unnoticed at first, the Internet and the Web emerged.

Today the PC is losing the spotlight. Organizations buy hundreds of embedded systems, sensors, and effectors, but these are weak devices with little memory, hard to network. Some tasks can be handed off to a second processor, but how far can parallel multicore computers take us? Government and industry are investing large sums in parallel computing. They are rediscovering the difficulties. Sensors and effectors will add processing and memory, harvest energy, and get networked. What will that lead to? The role of the PC may shift, becoming a personal control station where we can monitor vast quantities of information on anything of interest—our health, the state of household appliances, Internet activity, etc.—on large displays, with specific tasks easily moved to portable or distributed devices.

New technologies capture our attention, but of equal importance is the rapid maturation of technologies such as digital video and document repositories, as well as the complex specialization occurring in virtually all domains of application. Different patterns of use emerge in different cultures and different industries. Accessibility and sustainability are wide-open, specialized research and development areas. Tuning technologies for specific settings can bring human factors approaches to the fore; designing for efficient heavy use could revive command-driven interfaces, whether the commands are typed, spoken, or gestural.

Digital technology has inexorably increased the visibility of activity. We see people behaving not as we thought they would or as we think they should. Rules, conventions, policies, regulations, and laws are not consistently followed; sanctions for violating them are not uniformly applied. Privacy and our evolving attitudes toward it are a small piece of this powerful progression. Choosing how to approach these complex and intensifying challenges—Where do we increase enforcement? Should or could we create more nuanced rules? When should we tolerate more deviance?—at the levels of families, organizations, and societies. This will be a perpetual preoccupation as technology exposes the world as it is.

Until some time after it is revoked, Moore's law broadly construed will ensure that digital landscapes provide new forms of interaction to explore and new practices to improve. The first generation of computer researchers, designers, and users grew up without computers. The generation that followed used computers as students, entered workplaces, and changed the way technology was used. Now a generation has grown up with computers, game consoles, and cell phones. They absorbed an aesthetic of technology design and communicate by messaging. They are developing skills at searching, browsing, assessing, and synthesizing information. They use smartphones, acquire multimedia authoring talent, and embrace social networking sites. They have different takes on privacy and multitasking. They are entering workplaces, and everything will be changed once again. However it is defined and wherever it is studied, human–computer interaction will for some time be in its early days.

APPENDIX: PERSONAL OBSERVATIONS

My career from 1973 to 1993 followed a common enough path. I was one of many who worked as a computer programmer, studied cognitive psychology, spent time as an HCI professional in industry, and then moved to academia. I describe personal experiences here not because I am special, but to add texture and a sense of the human impact of some of the developments I have described. My interest in history arose from the feeling of being swept along by invisible forces, sometimes against my intention. My first effort at understanding was titled "The Computer Reaches Out" (Grudin 1990): I saw computers evolving and slowly reaching into the world and changing it in ways that we, their developers, had not foreseen.

1970: A Change in Plans

As a student, I read and believed the *Life* magazine article that forecast computers with superhuman intelligence arriving in several years. I concluded that if we survived a few years, we could count on machines to do all useful work. Human beings should focus on doing what they enjoy. I shifted from physics to mathematics and from politics to literature.

1973: Three Professions

Looking for my first job in 1973, I found three computer job categories in the *Boston Globe* classifieds: (1) operators, (2) programmers, and (3) systems analysts. Not qualified to be a highly paid analyst, I considered low-paid, hands-on operator jobs, but I landed a programming job with Wang Laboratories, which was at the time a small electronics company. For 2 years, I never saw the computer that my programs ran on. I flowcharted on paper and coded on coding sheets that a secretary sent to be punched and verified. A van carried the stack of cards 20 miles to a computer center, and later that day or the next day I got the printout. It might say something like "Error in Line 20," and I would resume work on the program.

1975: A Cadre of Discretionary Hand-On Users

In 1975, Wang acquired a few teletype terminals with access to the WYLBUR line editor, developed at the Stanford Linear Accelerator. Some of us programmers chose to abandon paper and became hands-on computer users.

1983: Chilly Reception for a Paper on Discretion in Use

My first HCI publication, Grudin and MacLean (1984), was written when I was a postdoctoral researcher at the MRC Applied Psychology Unit. Allan and I showed that people sometimes choose a slower interface for aesthetic or other reasons even when they are familiar with a more efficient alternative. A senior colleague asked us not to publish it. He worked on improving expert efficiency through cognitive modeling. A demonstration that greater efficiency could be undesirable would be a distraction, he said: "Sometimes the larger enterprise is more important than a small study."

1984: Encountering Moore's Law, Information Systems, Human Factors, and Design

I returned to Wang, which had become a large minicomputer company, and found that Moore's law had changed the business. More hardware was ordered from catalogs and the reduced cost of memory and other factors had changed programming priorities and skills. I was soon influenced by another cognitive psychologist, Susan Ehrlich, who worked in a marketing research group and later managed the human factors group. She introduced me to the IS literature, which I found difficult to understand. I attended Boston-area chapter meetings of both HFS and SIGCHI. I saw the cultural differences but felt CHI could learn from human factors. In a futile gesture to counter CHI antipathy toward human factors, I began calling myself a human factors engineer. I drove to Cambridge to see the newly released Macintosh. Few software engineers had the visual design skills that I realized would become important, so at work I looked for industrial designers of hardware (boxes) who could be enlisted to support software interface design.

1985: The Graphical User Interface Shock

In the early 1980s, Phil Barnard and I were among the many cognitive psychologists working on command naming. This was an important application in the era of command-line interfaces, but the ambition was to develop a comprehensive theoretical foundation for HCI. The success of the Mac in 1985 curtailed interest in command names. No one would build on our past work—a depressing thought. It also dashed the hope for a comprehensive theoretical foundation for HCI. We had to choose: Am I a cognitive psychologist or a computer professional? Phil remained a psychologist.

1986: Beyond the User: Groups and Organizations

I agreed to join MCC, an industry research consortium. Between jobs I worked on two papers, each addressing a major challenge encountered in product development: (1) From 1984 to 1986, I had worked on several products or features intended to support groups rather than individual users. These had not done well. Why was group support so challenging? (2) It was painfully evident that organizational structures and development processes were badly suited to interactive software development. What could be done about it? These issues formed the basis for much of my subsequent research.

1989: Development Contexts: A Major Differentiator

I spent 2 years at Aarhus University. Within weeks of arriving in a country that had little commercial software development, I saw that differences in the conditions that govern product, in-house, and contract development of interactive software could shape practices and perceptions in CHI, IS, and software engineering. Sorting this out led to my first library research for purely historical purposes (Grudin 1991). Perusing long-forgotten journals and magazines in dusty library corridors felt like wandering through an archaeological site.

1990: Just Words: Terminology Can Matter

I felt a premonition in 1987 when my IS-oriented colleague Susan Ehrlich titled a paper “Successful Implementation of Office Communication Systems.” By “implementation,” she meant introduction into organizations. To me, implementation was a synonym for coding or development. Sure enough, the ACM editor asked her to change the word implementation to adoption (Ehrlich 1987). What she called systems I called applications. Was language, usually an ally, getting in the way?

In 1990, I described the focus of my planned HCI course at Aarhus as “user-interface evaluation.” My new colleagues seemed embarrassed. Weeks later, a book written by one of them was published (Bødker 1990). Its first sentence was a quotation: “Design is where the action is, not evaluation.” Now I was embarrassed. In an in-house development world, with its dogma of getting the design right up front, development projects could take 10 years. Evaluation occurred at the end when only cosmetic changes were possible, and had a negative stigma. In commercial product development, evaluation of the previous version, competitive products, and (ideally) prototypes was integral to design. Evaluation is central to iterative design. It draws on the experimental psychologists’ skillset. We considered it a good thing.

Later in 1990, I participated in a panel on task analysis at a European conference. To my dismay, this IS-oriented group defined task analysis differently than I did. To them, it meant an organizational task analysis: tasks as components in a broad work process. In CHI, it meant a cognitive task analysis: breaking a simple task into components; for example, is “move text” thought of as “select-delete-paste” or as “select-move-place”? Some Europeans felt North American claims to have conducted task analyses were disgraceful, not understanding the context.

Also in 1990, en route to giving a job talk at the University of California Irvine, my first lecture to an IS audience at the University of California Los Angeles Anderson School of Management ended badly when the department head asked a question. It seemed meaningless, so I replied cautiously. He rephrased the question. I rephrased my response. He started again, then stopped and shrugged as if to say, “this fellow is hopeless.” When I saw him a few months later, he was astonished to learn that his Irvine friends were hiring me. Later, I understood the basis of our failure to communicate: We attached different meanings to the word “users.” To me, it meant hands-on computer users. He was asking about IS users who specified database requirements and read reports, but were not hands-on computer users. To me all use was hands-on, so his question had made no sense.

A book could be written about the word “user.” From a CHI perspective, the IS user was “the customer.” Consultants use “client.” In IS, the hands-on user was “the end-user.” In CHI parlance, end-user and user were one and the same—a person who entered data and used the output. The word end-user seemed superfluous or an affectation. Human factors used "operator" which CHI considered demeaning. In software engineering, user typically denoted a tool user, that is, a software engineer.

A final terminology note: the male generic. I avoided submitting to *IJMMS* and turned down an invitation to speak at a "man–machine interaction" event. I was keen on learning from other disciplines, but that was a linguistic bridge I usually avoided crossing. I generally consider words to be a necessary but uninteresting medium for conveying meaning, but such experiences led to an essay on unintended consequences of language (Grudin 1993).

2010: Reflections on Bridging Efforts

I have been a minor participant in efforts to find synergies drawing from CHI and HFS, OIS, IS (in both CSCW and AIS SIGHCI), or Design. None succeeded. I've interviewed others who participated years ago and identified the obstacles touched on in the introduction, many of which I experienced. As a boomer, I experienced generational and cultural divides. Many of my MCC colleagues joined the consortium to avoid Star Wars military projects. We lived through disputes between cognitive psychologists and radical behaviorists. I was among CHI researchers who shifted from journals to conferences as the primary publication venue and from hypothesis-driven experimentation to build-and-assess and qualitative field research.

Some differences fade over time, but many persist. Conference reviewers are often irritated by unfamiliar acronyms used by authors from other fields. Writing a chapter for an IS-oriented book (Palen and Grudin 2002), my coauthor and I wrangled at great length with the editor over terminology.

In researching this article, I reviewed the literature on TAM, the model of white-collar employee perceptions of technology that is heavily cited in IS but never in CHI. I unsuccessfully searched online for TAM references. Only on my third attempt did I see the problem: TAM stands for "Technology Acceptance Model," but I repeatedly typed in "Technology Adoption Model." TAM examined nondiscretionary acceptance, I think in terms of discretionary adoption. Different biases lead to different terminology, and confusion.

2010: Predicting the Future

Detailed forecasts, including mine, rarely look good upon close inspection. But understanding the forces that have shaped the past offers hope of anticipating or reacting quickly to future events. Even more useful may be indications of where effort will be futile. I believe the most common error is to underestimate the impact of hardware changes, and in particular that once effects start to be felt, how rapidly they will escalate. I published some analysis and projection in the November 2006 and January 2007 issues of *ACM Interactions*—check to see how I'm doing. (http://interactions.acm.org/content/archives.php).

REFERENCES*

Ackoff, R. L. 1967. Management misinformation systems. *Manage Sci* 14:B147–56.

Asimov, I. 1950. *I, Robot*. New York: Gnome Press.

Aspray, W. 1999. Command and control, documentation, and library science: The origins of information science at the University of Pittsburgh. *IEEE Ann Hist Comput* 21(4):4–20.

Baecker, R., and W. Buxton. 1987. A historical and intellectual perspective. In *Readings in HCI: A multidisciplinary approach*, ed. R. Baecker and W. Buxton, 41–54. San Francisco, CA: Morgan Kaufmann.

Baecker, R., J. Grudin, W. Buxton, and S. Greenberg. 1995. A historical and intellectual perspective. In *Readings in HCI: Toward the Year 2000*, ed. R. Baecker, J. Grudin, W. Buxton and S. Greenberg, 35–47. San Francisco, CA: Morgan Kaufmann.

Bagozzi, R. P., F. D. Davis, and P. R. Warshaw. 1992. Development and test of a theory of technological learning and usage. *Hum Relat* 45(7):660–86.

Banker, R. D., and R. J. Kaufmann. 2004. The evolution of research on Information Systems: A fiftieth-year survey of the literature in Management Science. *Manage Sci* 50(3):281–98.

Bannon, L. 1991. From human factors to human actors: The role of psychology and HCI studies in system design. In *Design at Work*, ed. J. Greenbaum and M. Kyng, 25–44. Hillsdale, NJ: Erlbaum.

Bardini, T. 2000. *Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing*. Stanford, CA: Stanford University.

Barnard, P. 1991. Bridging between basic theories and the artifacts of HCI. In *Designing Interaction: Psychology at the Human-Computer Interface*, ed. J. M. Carroll, 103–27. Cambridge: Cambridge University Press.

Barnard, P., J. May, D. Duke, and D. Duce. 2000. Systems, interactions, and macrotheory. *ACM Trans Comput Hum Interact* 7(2):222–62.

Begeman, M., P. Cook, C. Ellis, M. Graf, G. Rein, and T. Smith. 1986. Project Nick: Meetings augmentation and analysis. In *Proceedings Computer-Supported Cooperative Work 1986*, 1–6. Austin, TX: CSCW '86.

Benbasat, I., and A. S. Dexter. 1985. An experimental evaluation of graphical and color-enhanced information presentation. *Manage Sci* 31(11):1348–64.

Bennett, J. L. 1979. The commercial impact of usability in interactive systems. In *Man-computer communication*, ed. B. Shackel, Vol. 2, 1–17. Maidenhead, UK: Pergamon-Infotech.

Bewley, W. L., T. L. Roberts, D. Schroit, and W. L. Verplank. 1983. Human factors testing in the design of Xerox's 8010 "Star" office workstation. In *Proceedings CHI'83*, 72–77. New York: ACM.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design—Defining Customer-Centered Systems*. San Francisco, CA: Morgan Kaufmann.

Bjerknes, G., P. Ehn, and M. Kyng, eds. 1987. *Computers and Democracy—a Scandinavian Challenge*. Aldershot, UK: Avebury.

Björn-Andersen, N., and B. Hedberg. 1977. Design of information systems in an organizational perspective. In *Prescriptive Models of Organizations*, ed. P. C. Nystrom and W. H. Starbuck, Vol. 5, 125–42. *TIMS Studies in the Management Sciences*. Amsterdam, Netherlands: North-Holland.

* All URLs were accessed December 9, 2011.

Blackwell, A. 2006. The reification of metaphor as a design tool. *ACM Trans Comput Hum Interact* 13(4):490–530.

Blythe, M. A., A. F. Monk, K. Overbeeke, and P. C. Wright, eds. 2003. *Funology: From Usability to User Enjoyment*. New York: Kluwer.

Borman, L. 1996. SIGCHI: The early years. *SIGCHI Bull* 28(1):1–33. New York: ACM.

Borman, L. ed. 1981. *Proceedings of the Joint Conference on Easier and More Productive Use of Computer Systems, Part II: Human interface and user interface*. New York: ACM.

Bowman, W. J. 1968. *Graphic Communication*. New York: John Wiley.

Buckland, M. 1998. Documentation, information science, and library science in the U.S.A. In *Historical Studies in Information Science*, ed. T. B. Hahn, and M. Buckland, 159–72. Medford, NJ: Information Today/ASIS.

Buckland, M. 2009. As we may recall: Four forgotten pioneers. *Interactions* 16(6):76–69.

Burke, C. 1994. *Information and Secrecy: Vannevar Bush, Ultra, and the Other Memex*. Lanham, MD: Scarecrow Press.

Burke, C. 1998. A rough road to the information highway: Project INTREX. In *Historical studies in Information Science*, ed. T. B. Hahn and M. Buckland, 132–46. Medford, NJ: Information Today/ASIS.

Burke, C. 2007. History of information science. In *Annual review of Information Science and Technology 41*, ed. B. Cronin, 3–53. Medford, NJ: Information Today/ASIST.

Bush, V. 1945. As we may think. *Atl Mon* 176:101–8. http://www.theatlantic.com/magazine/archive/1969/12/as-we-may-think/3881/.

Butler, P. 1933. *Introduction to Library Science*. Chicago, IL: Univ. of Chicago Press.

Buxton, W. A. S. 2006. Early interactive graphics at MIT Lincoln Labs. http://www.billbuxton.com/Lincoln.html

Bødker, S. 1990. *Through the Interface: A Human Activity Approach to User Interface Design*. Mahwah, NJ: Lawrence Erlbaum.

Cakir, A., D. J. Hart, and T. F. M. Stewart. 1980. *Visual Display Terminals*. New York: Wiley.

Card, S. K., and T. P. Moran. 1986. User technology: From pointing to pondering. In *Proceedings of the Conference on the History of Personal Workstations*, 183–98. New York: ACM.

Card, S. K., T. P. Moran, and A. Newell. 1980a. Computer text-editing: An information-processing analysis of a routine cognitive skill. *Cogn Psychol* 12:396–410.

Card, S. K., T. P. Moran, and A. Newell. 1980b. Keystroke-level model for user performance time with interactive systems. *Commun ACM* 23(7):396–410. New York: ACM.

Card, S., T. P. Moran, and A. Newell. 1983. The psychology of human-computer interaction. Mahwah, NJ: Lawrence Erlbaum Associates.

Carey, J. 1988. *Human Factors in Management Information Systems*. Greenwich, CT: Ablex.

Carroll, J. M., ed. 2003. *HCI Models, Theories and Frameworks: Toward a Multidisciplinary Science*. San Francisco, CA: Morgan Kaufmann.

Carroll, J. M., and R. L. Campbell. 1986. Softening up hard science: Response to Newell and Card. *Hum Comput Interact* 2(3):227–49.

Carroll, J. M., and S. A. Mazur. 1986. Lisa learning. *IEEE Comput* 19(11):35–49.

Cronin, B. 1995. Shibboleth and substance in North American Library and Information Science education. *Libri* 45:45–63.

Damodaran, L., A. Simpson, and P. Wilson. 1980. *Designing Systems for People*. Manchester, UK: NCC Publications.

Darrach, B. 1970. Meet Shaky: The first electronic person. *Life Mag* 69(21):58B–68.

Davis, F. D. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q* 13(3):319–39.

Davis, G. B. 1974. *Management Information Systems: Conceptual Foundations, Structure, and Development*. New York: McGraw-Hill.

Dennis, A., J. George, L. Jessup, J. Nunamaker, and D. Vogel. 1988. Information technology to support electronic meetings. *MIS Q* 12(4):591–624.

DeSanctis, G., and R. B. Gallupe. 1987. A foundation for the study of group decision support systems. *Manage Sci* 33:589–610.

Dyson, F. 1979. *Disturbing the Universe*. New York: Harper and Row.

Ehrlich, S. F. 1987. Strategies for encouraging successful adoption of office communication systems. *ACM Trans Office Inf Syst* 5(4):340–57.

Engelbart, D. 1962. Augmenting human intellect: A conceptual framework. SRI Summary Report AFOSR-3223. Reprinted in *Vistas in Information Handling*, ed. P. Howerton and D. Weeks, Vol. 1, 1–29. Washington, DC: Spartan Books, 1963. http://www.dougengelbart.org/pubs/augment-3906.html

Engelien, B., and R. McBryde. 1991. *Natural Language Markets: Commercial Strategies*. London: Ovum Ltd.

Evenson, S. 2005. Design and HCI highlights. Presented at the HCIC 2005 Conference, Winter Park, Colorado, February 6, 2005.

Fano, R., and F. Corbato. 1966. Time-sharing on computers. *Sci Am* 214(9):129–40.

Feigenbaum, E. A., and P. McCorduck. 1983. *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*. Reading, MA: Addison-Wesley.

Fidel, R. 2011. *Human Information Interaction: An Ecological Approach to Information Behavior*. Cambridge, MA: MIT Press.

Foley, J. D., and V. L. Wallace. 1974. The art of natural graphic man-machine conversation. *Proc IEEE* 62(4):462–71.

Forbus, K. 2003. *Sketching for Knowledge Capture*. Lecture at Microsoft Research, Redmond, WA, May 2.

Forbus, K. D., J. Usher, and V. Chapman. 2003. Qualitative spatial reasoning about sketch maps. In *Proceedings of the Innovative Applications of AI*, 85–92. Menlo Park: AAAI.

Forster, E. M. 1909. *The Machine Stops*. Oxford and Cambridge Review, 8, November, 83–122.

Friedman, A. 1989. *Computer Systems Development: History, Organization and Implementation*. New York: Wiley.

Gilbreth, L. 1914. *The Psychology of Management: The Function of the Mind in Determining Teaching and Installing Methods of Least Waste*. NY: Sturgis and Walton.

Good, I. J. 1965. Speculations concerning the first ultra-intelligent machine. *Adv Comput* 6:31–88. http://commonsenseatheism.com/wp-content/uploads/2011/02/Good-Speculations-Concerning-the-First-Ultraintelligent-Machine.pdf.

Gould, J. D., and C. Lewis. 1983. Designing for usability—Key principles and what designers think. In *Proceedings of CHI'83*, 50–3. New York: ACM.

Grandjean, E., and A. Vigliani. 1980. *Ergonomics Aspects of Visual Display Terminals*. London: Taylor and Francis.

Gray, W. D., B. E. John, R. Stuart, D. Lawrence, and M. E. Atwood. 1990. GOMS meets the phone company: Analytic modeling applied to real-world problems. In *Proceedings of Interact'90*, 29–34. Amsterdam: North Holland.

Greenbaum, J. 1979. *In the Name of Efficiency*. Philadelphia, PA: Temple University.

Greif, I. 1985. Computer-Supported Cooperative Groups: What are the issues? In *Proceedings AFIPS Office Automation Conference*, 73–6. Montvale, NJ: AFIPS Press.

Greif, I., ed. 1988. *Computer-Supported Cooperative Work: A Book of Readings*. San Mateo, CA: Morgan Kaufmann.

Grudin, J. 1990. The computer reaches out: The historical continuity of interface design. In *Proceedings of CHI'90*, 261–8. New York: ACM.

Grudin, J. 1991. Interactive systems: Bridging the gaps between developers and users. *IEEE Comput* 24(4):59–69.

Grudin, J. 1993. Interface: An evolving concept. *Commun ACM* 36(4):110–9.

Grudin, J. 2009. AI and HCI: Two fields divided by a common focus. *AI Mag* 30(4):48–57.

Grudin, J. 2010. Conferences, community, and technology: Avoiding a crisis. In *Proceedings iConference 2010*. https://www.ideals.illinois.edu/handle/2142/14921.

Grudin, J. 2011. Human-computer interaction. In *Annual Review of Information Science and Technology*, ed. B. Cronin, Vol. 45, 369–430. Medford, NJ: Information Today (for ASIST).

Grudin, J., and A. MacLean. 1984. Adapting a psychophysical method to measure performance and preference tradeoffs in human-computer interaction. In *Proceedings of INTERACT'84*, 338–42. Amsterdam, Netherlands: North Holland.

Hertzfeld, A. 2005. *Revolution in the Valley: The Insanely Great Story of How the Mac Was Made*. Sebastopol, CA: O'Reilly Media.

HFES. 2010. HFES history. In *HFES 2010–2011, Directory and Yearbook*, 1–3. Santa Monica, CA: Human Factors and Ergonomics Society. Also found at http://www.hfes.org/web/AboutHFES/history.html

Hiltz, S. R., and M. Turoff. 1978. *The Network Nation*. Reading, MA: Addison-Wesley.

Hiltzik, M. A. 1999. *Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age*. New York: HarperCollins.

Hopper, G. 1952. The education of a computer. In *Proceedings of ACM Conference*, Reprinted in Annals of the History of Computing, 9(3–4), 271–81, 1987.

Huber, G. 1983. Cognitive style as a basis for MIS and DSS designs: Much ado about nothing? *Manage Sci* 29(5):567–79.

Hutchins, E. L., J. D. Hollan, and D. A. Norman. 1986. Direct manipulation interfaces. In *User Centered System Design*, ed. D. A. Norman and S. W. Draper, 87–124. Mahwah, NJ: Lawrence Erlbaum.

Israelski, E., and A. M. Lund. 2003. The evolution of HCI during the telecommunications revolution. In *The Human-Computer Interaction Handbook*, ed. J. A. Jacko and A. Sears, 772–89. Mahwah, NJ: Lawrence Erlbaum.

Johnson, T. 1985. *Natural Language Computing: The Commercial Applications*. London, UK: Ovum Ltd.

Kao, E. 1998. *The History of AI*. http://www.generation5.org/content/1999/aihistory.asp (accessed March 13, 2007).

Kay, A., and A. Goldberg. 1977. Personal dynamic media. *IEEE Comput* 10(3):31–42.

Keen, P. G. W. 1980. MIS research: Reference disciplines and a cumulative tradition. In *First International Conference on Information Systems*, 9–18. Chicago, IL: Society for Management Information Systems.

Kling, R. 1980. Social analyses of computing: Theoretical perspectives in recent empirical research. *Comput Surv* 12(1):61–110.

Landau, R., J. Bair, and J. Siegmna, eds. 1982. Emerging office systems. In *Extended Proceedings of the 1980 Stanford International Symposium on Office Automation*. Norwood, NJ.

Lenat, D. 1989. When will machines learn? *Mach Learn* 4:255–7.

Lewis, C. 1983. The 'thinking aloud' method in interface evaluation. Tutorial given at CHI'83. Unpublished notes.

Lewis, C., and R. Mack. 1982. Learning to use a text processing system: Evidence from "thinking aloud" protocols. In *Proceedings of the Conference on Human Factors in Computing Systems*, 387–92. New York: ACM.

Lewis, C., P. Polson, C. Wharton, and J. Rieman. 1990. Testing a walk-through methodology for theory-based design of walk-up-and-use Interfaces. In *Proceedings of the Conference on Human Factors in Computing Systems*, 235–42. New York: ACM.

Licklider, J. C. R. 1960. Man-computer symbiosis. *IRE Trans Hum Factors Electron HFE-1* 1:4–11. http://groups.csail.mit.edu/medg/people/psz/Licklider.html.

Licklider, J. C. R. 1963. MEMORANDUM FOR: Members and Affiliates of the Intergalactic Computer Network. April 23. http://www.kurzweilai.net/memorandum-for-members-and-affiliates-of-the-intergalactic-computer-network.

Licklider, J. C. R. 1965. *Libraries of the Future*. Cambridge, MA: MIT Press.

Licklider, J. C. R. 1976. User-oriented interactive computer graphics. In *Proceedings of SIGGRAPH Workshop on User-Oriented Design of Interactive Graphics Systems*, 89–96. New York: ACM.

Licklider, J. C. R., and W. Clark. 1962. On-line man-computer communication. *AFIPS Conf Proc* 21:113–28.

Lighthill, J. 1973. Artificial intelligence: A general survey. In *Artificial Intelligence: A Paper Symposium*, ed. J. Lighthill, N. S. Sutherland, R. M. Needham, H. C. Longuet-Higgins, and D. Michie. London, UK: Science Research Council of Great Britain. http://www.chilton-computing.org.uk/inf/literature/reports/lighthill_report/p001.htm

Long, J. 1989. Cognitive ergonomics and human-computer interaction. In *Cognitive Ergonomics and Human-Computer Interaction*, ed. J. Long and A. Whitefield, 4–34. Cambridge: Cambridge University Press.

Machlup, F., and U. Mansfield, eds. 1983. *The Study of Information: Interdisciplinary Messages*. New York: Wiley.

March, A. 1994. Usability: the new dimension of product design. *Harv Bus Rev* 72(5):144–9.

Marcus, A. 2004. Branding 101. *ACM Interact* 11(5):14–21.

Markoff, J. 2005. *What the Dormouse Said: How the 60s Counter-Culture Shaped the Personal Computer*. London, UK: Viking.

Markus, M. L. 1983. Power, politics, and MIS implementation. *Commun ACM* 26(6):430–44.

Martin, J. 1973. *Design of Man–Computer Dialogues*. New York: Prentice-Hall.

McCarthy, J. 1960. Functions of symbolic expressions and their computation by machine, part 1. *Commun ACM* 3(4):184–95.

McCarthy, J. 1988. B. P. Bloomfield, The question of artificial intelligence: Philosophical and sociological perspectives. *Ann Hist Comput* 10(3):224–9. http://www-formal.stanford.edu/jmc/reviews/bloomfield/bloomfield.html.

Meister, D. 1999. *The History of Human Factors and Ergonomics*. Mahwah, NJ: Lawrence Erlbaum.

Moggridge, B. 2007. *Designing Interactions*. Cambridge: MIT Press.

Moravec, H. 1998. When will computer hardware match the human brain? *J Evol Technol* 1:1. http://www.transhumanist.com/volume1/moravec.htm.

Mumford, E. 1971. A comprehensive method for handling the human problems of computer introduction. *IFIP Congr* 2:918–23.

Mumford, E. 1976. Toward the democratic design of work systems. *Personnel Manage* 8(9):32–5.

Myers, B. A. 1998. A brief history of human computer interaction technology. *ACM Interact* 5(2):44–54.

National Science Foundation. 1993. *NSF 93–2: Interactive Systems Program Description*. http://www.nsf.gov/pubs/stis1993/nsf932/nsf932.txt

National Science Foundation. 2003. NSF Committee of Visitors Report: Information and Intelligent Systems Division, 28 July 2003.

Negroponte, N. 1970. *The Architecture Machine: Towards a More Humane Environment*. Cambridge: MIT Press.

Nelson, T. 1965. A file structure for the complex, the changing, and the indeterminate. In *Proceedings of the ACM National Conference*, 84–100. New York: ACM.

Nelson, T. 1973. A conceptual framework for man-machine everything. In *Proceedings of the National Computer Conference*, M21–6. Montvale, NJ: AFIPS Press.

Nelson, T. 1996. Generalized links, micropayment and transcopyright. http://www.almaden.ibm.com/almaden/npuc97/1996/tnelson.htm.

Newell, A., and S. K. Card. 1985. The prospects for psychological science in human-computer interaction. *Hum Comput Interact* 1(3):209–42.

Newell, A., and H. A. Simon. 1956. The logic theory machine: A complex information processing system. *IRE Trans Inf Theory* IT-2:61–79.

Newell, A., and H. A. Simon. 1972. *Human Problem Solving*. New York: Prentice-Hall.

Newman, W. M., and R. F. Sproull. 1973. *Principles of Interactive Computer Graphics*. New York: McGraw-Hill.

Nielsen, J. 1989. Usability engineering at a discount. In *Designing and Using Human-Computer Interfaces and Knowledge Based Systems*, ed. G. Salvendy and M. J. Smith, 394–401. Amsterdam: Elsevier.

Nielsen, J., and R. Molich. 1990. Heuristic evaluation of user interfaces. In *Proceedings of CHI'90*, 249–56. New York: ACM.

Norberg, A. L., and J. E. O'Neill. 1996. *Transforming Computer Technology: Information Processing for the Pentagon*, 1962–86. Baltimore, MD: Johns Hopkins.

Norman, D. A. 1982. Steps toward a cognitive engineering: Design rules based on analyses of human error. In *Proceedings of the Conference on Human Factors in Computing Systems*, 378–82. New York: ACM.

Norman, D. A. 1983. Design principles for human-computer interfaces. In *Proc. CHI'83*, 1–10. New York: ACM.

Norman, D. A. 1986. Cognitive engineering. In *User Centered System Design*, ed. D. A. Norman and S. W. Draper, 31–61. Mahwah, NJ: Lawrence Erlbaum.

Norman, D. A. 1988. *Psychology of Everyday Things. Reissued as 'Design of Everyday Things' in 1990*. New York: Basic Books.

Norman, D. A. 2004. *Emotional Design: Why We Love (or hate) Everyday Things*. New York: Basic Books.

Nunamaker, J., R. O. Briggs, D. D. Mittleman, D. R. Vogel, and P. A. Balthazard. 1997. Lessons from a dozen years of group support systems research: A discussion of lab and field findings. *J Manage Inf Syst* 13(3):163–207.

Nygaard, K. 1977. Trade union participation. Presentation at CREST Conference on Management Information Systems, Stafford, UK.

Oakley, B. W. 1990. Intelligent knowledge-based systems—AI in the U.K. In *The Age of Intelligent Machines*, ed. R. Kurzweil, 346–9. Cambridge, MA: MIT Press.

Palen, L., and J. Grudin. 2002. Discretionary adoption of group support software. In *Implementing Collaboration Technology in Industry*, ed. B. E. Munkvold, 159–90. London, UK: Springer-Verlag.

Perlman, G., G. K. Green, and M. S. Wogalter. 1995. *Human Factors Perspectives on Human-Computer Interaction.* Santa Monica: Human Factors and Ergonomics Society.

Pew, R. 2003. Evolution of HCI: From MEMEX to Bluetooth and beyond. In *The Human-Computer Interaction Handbook*, ed. J. A. Jacko and A. Sears, 1–17. Mahwah, NJ: Lawrence Erlbaum.

Pruitt, J., and T. Adlin. 2006. *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*. San Francisco, CA: Morgan Kaufmann.

Rasmussen, J. 1980. The human as a system component. In *Human Interaction with Computers*, ed. H. T. Smith and T. R. G. Green, 67–96. London, UK: Academic.

Rasmussen, J. 1986. *Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering*. New York: North-Holland.

Rayward, W. B. 1983. Library and information sciences: Disciplinary differentiation, competition, and convergence. In *The Study of Information: Interdisciplinary Messages*, ed. F. Machlup and U. Mansfield, 343–405. New York: Wiley.

Rayward, W. B. 1998. The history and historiography of Information Science: Some reflections. In *Historical Studies in Information Science*, ed. T. B. Hahn, and M. Buckland, 7–21. Medford, NJ: Information Today/ASIS.

Remus, W. 1984. An empirical evaluation of the impact of graphical and tabular presentations on decision-making. *Manage Sci* 30(5):533–42.

Roscoe, S. N. 1997. *The Adolescence of Engineering Psychology*. Santa Monica, CA: Human Factors and Ergonomics Society.

Sammet, J. 1992. Farewell to Grace Hopper—End of an era! *Commun ACM* 35(4):128–31.

Shackel, B. 1959. Ergonomics for a computer. *Design* 120:36–9.

Shackel, B. 1962. Ergonomics in the design of a large digital computer console. *Ergonomics* 5:229–41.

Shackel, B. 1997. HCI: Whence and whither? *J ASIS* 48(11):970–86.

Shannon, C. E. 1950. Programming a computer for playing chess. *Philos mag* 7(41):256–75.

Shannon, C. E., and W. Weaver. 1949. *The Mathematical Theory of Communication*. Urbana, IL: Univ. of Illinois Press.

Sheil, B. A. 1981. The psychological study of programming. *ACM Comput Surv* 13(1):101–20.

Shneiderman, B. 1980. *Software Psychology: Human Factors in Computer and Information Systems*. Cambridge, MA: Winthrop.

Simon, H. A. 1960. *The New Science of Management Decision*. New York: Harper.

Smith, H. T., and T. R. G. Green, eds. 1980. *Human Interaction with Computers*. Orlando, FL: Academic.

Smith, S. L. 1963. Man-computer information transfer. In *Electronic Information Display Systems*, ed. J. H. Howard, 284–99. Washington, DC: Spartan Books.

Smith, S. L., B. B. Farquhar, and D. W. Thomas. 1965. Color coding in formatted displays. *J Appl Psychol* 49:393–8.

Smith, S. L., and N. C. Goodwin. 1970. Computer-generated speech and man-computer interaction. *Hum Factors* 12:215–23.

Smith, S. L., and J. N. Mosier. 1986. *Guidelines for Designing User Interface Software (ESD-TR-86-278)*. Bedford, MA: MITRE.

Suchman, L. 1987a. *Plans and Situated Action: The Problem of Human-Machine Communication*. Cambridge, UK: Cambridge University Press.

Suchman, L. 1987b. Designing with the user: Review of 'Computers and democracy: A Scandinavian challenge. *ACM TOIS* 6(2):173–83.

Sutherland, I. 1963. *Sketchpad: A Man-Machine Graphical Communication System*. Doctoral Dissertation, MIT. http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-574.pdf

Taylor, F. W. 1911. *The Principles of Scientific Management*. New York: Harper.

Turing, A. 1949. *Letter in London Times*, June 11. See Highlights from the Computer Museum report Vol. 20, Summer/Fall 1987. http://ed-thelen.org/comp-hist/TCMR-V20.pdf.

Turing, A. 1950. Computing machinery and intelligence. *Mind* 49:433–60. Republished as "Can a machine think?" in J. R. Newman (Ed.), *The World of Mathematics*, (Vol. 4, pp. 2099–2123). New York: Simon and Schuster.

Vessey, I., and D. Galletta. 1991. Cognitive fit: An empirical test of information acquisition. *Inf Syst Res* 2(1):63–84.

Waldrop, M. M. 2001. *The Dream Machine: J.C.R. Licklider and the Revolution that Made Computing Personal*. New York: Viking.

Weinberg, G. 1971. *The Psychology of Computer Programming*. New York: Van Nostrand Reinhold.

Wells, H. G. 1905. *A Modern Utopia*. London, UK: Jonathan Cape. http://www.gutenberg.org/etext/6424.

Wells, H. G. 1938. *World Brain.* London, UK: Methuen.

Wersig, G. 1992. Information science and theory: A weaver bird's perspective. In *Conceptions of Library and Information Science: Historical, Empirical, and Theoretical Perspectives*, ed. P. Vakkari and B. Cronin, 201–17. London, UK: Taylor Graham.

White, P. T. 1970. Behold the computer revolution. *National Geographic* 38(5): 593–633. http://blog.modernmechanix.com/2008/12/22/behold-the-computer-revolution/

Yates, J. 1989. *Control Through Communication: The Rise of System in American Management.* Baltimore, MD: Johns Hopkins.

Zhang, P. 2004. AIS SIGHCI three-year report. *SIGHCI newslett* 3(1):2–6.

Zhang, P., F. F. -H. Nah, and J. Preece. 2004. HCI studies in management information systems. *Behav Inf Technol* 23(3):147–51.

Zhang, P., Li, N., Scialdone, M.J. & Carey, J. 2009. The intellectual advancement of Human-Computer Interaction Research: A critical assessment of the MIS Literature. Trans Human-Computer Interaction, 1(3): 55–107.

Part I

Humans in HCI

1 Perceptual-Motor Interaction
Some Implications for Human–Computer Interaction

Timothy N. Welsh, Sanjay Chandrasekharan, Matthew Ray, Heather Neyedli, Romeo Chua, and Daniel J. Weeks

CONTENTS

1.1 PERCEPTUAL-MOTOR INTERACTION: A BEHAVIORAL EMPHASIS

Human–computer interaction is going through a period of rapid evolution. Although mouse, keyboard, and joystick devices will continue to dominate for the immediate future, embodied, gestural, and tangible interfaces—where individuals use their body to directly manipulate information objects—are rapidly changing the computing landscape. Most new laptops and mobile devices now support multitouch, which allows us to use our fingers and gestures to directly manipulate virtual objects on the screen. Hence, as an alternative to pointing and clicking with a mouse, we can now directly pull, push, grab, pinch, squeeze, crush, and throw virtual objects. We can shake our portable music player to change the song we are listening to, or we can turn our mobile devices horizontally to get a wider display screen. Using the Wii Remote, we can now use our body movements to interact with objects in video games and manipulate them. Tangible and augmented interfaces now allow us to interact directly with virtual environments by moving actual objects on a tabletop (Hornecker et al. 2008). In sum, instead of being forced to use dissociated (mouse) and/or arbitrary (keyboard and joystick) sensorimotor mappings to achieve our goals, these new modes of interaction allow for a more direct mapping of our movements on to the work space. The "naturalness" and ease of operation of these interfaces are, in large part, due to the sensory and motor systems' close connection to cognition. Therefore, as these new interfaces become more popular, it is becoming increasingly important to consider the mechanisms that support such interactions.

In our studies of human–computer interaction (HCI) and perceptual-motor interactions in general, we have adopted a number of theoretical and analytical frameworks as part of an integrated approach. Our chapters in earlier editions of this handbook (Chua, Weeks, and Goodman 2003; Welsh et al. 2007) reviewed much of this research and its implications for HCI. The emphasis for these earlier chapters was on using information-processing approaches to understand the translation of perceptual into motor space and the interaction between processes of attention and action planning. Although our research has continued to explore the interplay between the processes of action and attention that we introduced in our chapter in the second edition (Welsh et al. 2007), we are intrigued by the possibilities offered by recently developed tangible interfaces and how theories of embodied cognition and common coding can support and enhance the progress of these systems.

Thus, in the present chapter, we have provided an updated review and expansion of our recent work in the area of action-centered attention and suggest some important implications for the role that action planning plays in the capture of attention and perception. We believe this work has important implications for the design of interaction modes. In the second section, we review the critical features of an alternative theoretical approach to cognition and action that presupposes an in-depth interaction between perception, cognition, and action. This latter theory has shaped much of our more recent work on the development of a tangible and embodied HCI. The critical theme that binds the seemingly diverse lines of work is the role that action planning has in information-processing systems. It is this central consideration that we argue has been lacking over the years and should be an important consideration for the future work.

1.1.1 Human Information Processing and Perceptual-Motor Behavior

The information-processing framework has traditionally provided a major theoretical and empirical platform for many scientists interested in perceptual-motor behavior. The study of perceptual-motor behavior within this framework has inquired into such issues as the information capacity of the motor system (e.g., Fitts 1954), the attentional demands of movements (e.g., Posner and Keele 1969), motor memory (e.g., Adams and Dijkstra 1966), and processes of motor learning (e.g., Adams 1971). The language of information processing (e.g., Broadbent 1958) has provided the vehicle for discussions of mental and computational operations of the cognitive and perceptual-motor system (Posner 1982). Of interest in the study of perceptual-motor behavior is the nature of the cognitive processes that underlie perception and action.

The information-processing approach describes the human as an active processor of information, in terms that are now commonly used to describe complex computing mechanisms. An information-processing analysis describes observed behavior in terms of the encoding of perceptual information, the manner in which internal psychological subsystems utilize the encoded information, and the functional organization of these subsystems. At the heart of the human cognitive system are processes of information transmission, translation, reduction, collation, storage, and retrieval (e.g., Fitts 1964; Marteniuk 1976; Stelmach 1982; Welford 1968). Consistent with a general model of human information processing (e.g., Fitts and Posner 1967), three basic processes have been distinguished historically. For our purposes, we refer to these processes as stimulus identification, response selection, and response programming. Briefly, stimulus identification is associated with processes responsible for the perception of information. Response selection pertains to the translation between stimuli and responses and the selection of a response. Response programming is associated with the organization of the final output (see Proctor and Vu 2003 or the present volume).

A key feature of early models of information processing is the emphasis upon the cognitive activities that precede action (Marteniuk 1976; Stelmach 1982). From this perspective, action is viewed only as the end-result of a complex chain of information-processing activities (Marteniuk 1976). Thus, chronometric measures such as reaction time and movement time, as well as other global outcome measures, are often the predominant dependent measures. However, even a cursory examination of the literature indicates that the time to engage a target has been a primary measure of interest. For example, a classic assessment of perceptual-motor behavior in the context of HCI and input devices was conducted by Card et al. (1978); see also English, Engelhart, and Berman (1967). Employing measures of error and speed, Card et al. (1978) had subjects complete a cursor-positioning task using four different control devices (mouse, joystick, step keys, and text keys). The data revealed the now well-known advantage for the mouse. Of interest is that the speed measure was decomposed into "homing" time, the time that it took to engage the control device and initiate cursor movement, and "positioning" time, the time to complete the cursor movement. Although the mouse was actually the poorest device in terms of the homing time measure, the advantage in positioning time produced the faster overall time. That these researchers sought to glean more information from the time measure acknowledges the importance of the movement itself in perceptual-motor interactions such as these.

The fact that various pointing devices depend on hand movement to control cursory movement has led researchers in HCI to emphasize Fitts's law (Fitts 1954) as a predictive model of time to engage a target. The law predicts pointing (movement) time as a function of the distance to and the width of the target—where, in order to maintain a given level of accuracy, movement time must increase as the distance of the movement increases and/or the width of the target decreases. The impact of Fitts's law is most evident by its inclusion in the battery of tests to evaluate computer pointing devices in ISO 9241-9. We argue that there are a number of important limitations to an exclusive reliance on Fitts's law in this context.

First, although the law predicts movement time, it does so on the basis of distance and target size. Consequently, it does not allow for determining what other factors may influence movement time. Specifically, Fitts's law is often based on a movement to a single target at any given time (although it was originally developed using reciprocal movements between two targets). However, in most HCI and graphical user interface contexts, there is an array of potential targets that can be engaged by an operator. These nontarget, but action-relevant stimuli in the movement environment can have profound and unexpected effects on action planning and execution. For example, Adam et al. (2006) have repeatedly found that the last target in an array enjoys a movement time advantage that is not predicted by Fitts's law. In contrast, distracting nontarget stimuli that capture attention can negatively affect both the temporal and physical characteristics of the movements to the imperative target. We will discuss these negative consequences in greater detail in Section 1.2.1.3.

Second, we suggest that the emphasis on Fitts's law has diverted attention from the fact that cognitive processes involving the selection of a potential target from an array are an important, and time-consuming, information-processing activity that must precede movement to that target. For example, the Hick–Hyman law (Hick 1952; Hyman 1953) predicts the decision time required to select a target response from a set of potential responses—where the amount of time required to choose the correct response increases with the number of possible alternative responses. What is important to understand is that the two laws work independently to determine the total time it takes for an operator to acquire the desired location. In one instance, an operator may choose to complete the decision making and movement components sequentially. Under these conditions, the total time to complete the task will be the sum of the times predicted by the Hick–Hyman and Fitts's laws. Alternatively, an operator may opt to make a general movement that is an approximate average of the possible responses and then select the final target destination while the movement is being completed. Under such conditions, Hoffman and Lim (1997) reported interference between the decision and movement component that was dependent on their respective difficulties (see also Meegan and Tipper 1998).

Finally, although Fitts's law predicts movement time given a set of movement parameters, it does not actually reveal much about the underlying movement itself. Indeed, considerable research effort has been directed toward revealing the movement processes that give rise to Fitts's law. For example, theoretical models of limb control have been forwarded that propose that Fitts's law emerges as a result of multiple submovements (e.g., Crossman and Goodeve 1963/1983), or as a function of both initial movement impulse variability and subsequent corrective processes late in the movement (Meyer et al. 1988). These models highlight the importance of conducting detailed examinations of movements themselves as a necessary complement to chronometric explorations.

For these reasons, HCI situations that involve dynamic perceptual-motor interactions may not be best indexed merely by chronometric methods (cf. Card et al. 1978). Indeed, as HCI moves beyond the simple key press interfaces that are characteristic of early systems to include virtual and augmented reality, teleoperation, gestural and haptic interfaces, among others, the dynamic nature of perceptual-motor interactions are even more evident. Consequently, assessment of the actual movement required to engage such interfaces will be more revealing.

To supplement chronometric explorations of basic perceptual-motor interactions, motor behavior researchers have also advocated a "movement process" approach (Kelso 1982). The argument is that in order to understand the nature of movement organization and control, analyses should also encompass the movement itself, and not just the activities preceding it (e.g., Kelso 1982, 1995; Marteniuk, MacKenzie, and Leavitt 1988). Thus, investigators have examined the kinematics of movements in attempts to further understand the underlying organization involved (e.g., Brooks 1974; Chua and Elliott 1993; Elliott et al. 1991; Kelso, Southard, and Goodman 1979; MacKenzie et al. 1987; Marteniuk et al. 1987). The relevance of this approach will become apparent in later sections.

1.1.2 Sensory Information during the Planning and Control of Action

It almost goes without saying that different types of actions need different types and amounts of information to ensure accurate completion. Theoretical and experimental considerations of this issue, in a manner that is relevant to the field of HCI, have been expanded recently. Before discussing the evidence supporting this view and outlining some potential implications for HCI, we will briefly review the processes involved in the planning and control of action and the types of information used during these processes. Readers interested in gaining a more in-depth understanding of this research should consult a recent book on the topic (Elliott and Khan 2010).

Since the seminal work of Woodworth (1899), it has been generally accepted that goal-directed action consists of two main components: (1) the ballistic or open-loop component that initiates the action toward the goal and (2) the current control or closed-loop component during which movement-produced information is used to facilitate movement accuracy. The initial open-loop component is thought to represent the results of the stages of information processing and initial plan or motor program the individual has developed to complete the goal successfully. The second component of the action begins after the movement has been initiated and directed toward the goal. During this part of the movement, sources of movement-produced information about the current location and trajectory of the effector (feedback) are compared with the predicted or desired location and trajectory to determine any differences between the actual and desired movement pattern (i.e., movement error). These error signals are then used to correct the unfolding movement and achieve the goal.

The main evidence in favor of the notion of planning and control components for goal-directed actions is derived from detailed analyses of the kinematic profiles of aiming actions performed under various stimulus conditions (e.g., Chua and Elliott 1993; Heath 2005; see Khan et al. 2006 for a review).

Although vestibular and proprioceptive information is also necessary for the accurate planning and control of movement, visual information is by far the dominant source and, as such, is the source of information that is most commonly manipulated in these studies. As one would expect, people are more accurate and less variable under conditions in which they have vision of the environment than when they do not. The increase in accuracy is thought to occur, in large part, because the actor has visual information of both the effector and the target to detect and correct errors in the trajectory.

Of greater importance to the present discussion, however, are the results of the in-depth kinematic analysis of the aiming movements. The consistent finding of this research is that the majority of the differences between the movements executed with and without vision appears in the later portions of the movement. Specifically, the initial segments of movements performed both with and without vision are characterized by relatively similar smooth increases and then decreases in velocity (bell-shaped profiles). It is thought that this relative consistency arises because there is a relative consistency in the motor programs that are the basis of these early portions of the movement in both vision and no vision conditions. In contrast to the similarities in the initial parts of the movements, the later portions of the movement performed with continuous visual information of the environment are characterized by a much larger number of sudden decelerations and reaccelerations than movements executed in the absence of visual information. These discontinuities in the kinematic profiles are thought to represent instances in which the actor has used visual information about the effector and the target to detect errors and then formulate and execute corrective submovements. These online corrections increase the accuracy of the movement. In the absence of vision, most errors go undetected leading to smoother deceleration phases (i.e., with fewer corrective submovements) and more end point error.

It is important to note here that not all actions consist of both components. Although each action needs a ballistic component to get the action initiated, actions may be successfully completed in the absence of feedback-based control. There are two common circumstances in which actions are completed without (or with minimal influence from) feedback-based control. The first circumstance in which feedback-based control is not needed is situations in which end point accuracy demands are minimal (e.g., when there is a low index of difficulty, the target is really large and/or close to the effector). Feedback-based corrections might not occur here because the programmed component of the action is accurate enough to achieve the goal. The second circumstance involves situations in which actions are completed in a very short amount of time. Because the feedback loops require time to effectively influence the actions, feedback-based corrections during rapid or ballistic actions are simply not possible. The actor still receives the response-produced information at the end of the movement and can determine whether they have successfully completed the response and can use that information to adjust the next action (i.e., make an offline correction to the action). The information, however, cannot be used online (during the action) to ensure its accurate completion. Thus, for ballistic actions, such as key presses, a continual source of target information during execution will not affect performance because online corrections cannot be made. Successful completion of action in this context is dependent on the accuracy of the motor program. In contrast, for movements with a longer execution time, such as finger- or mouse-based aiming movements, a continual source of information facilitates accurate completion because the information can be used to make online corrections to the unfolding action.

In sum, the critical implication from this discussion of the use of visual information in motor programming and control is that different types of actions require different types and amounts of information. Specifically, because key press responses are completed in a ballistic manner without the use of feedback, the stable sources of information regarding the target location are not needed to ensure accurate completion. In contrast, because aiming movements generally take longer to complete and have higher accuracy demands, a continual and stable source of visual information about the effector and the target is needed for efficient feedback-based corrections and movement accuracy. As will be discussed later, recent findings suggest that the ways in which we perceive and attend to objects in the world is determined, in part, by the to-be-performed response mode. Thus, careful consideration of response mode is necessary when designing work environment to ensure the efficient extraction of the relevant information and use of the system.

1.1.3 Translation, Coding, and Mapping

As outlined in the preceding sections, the dominant models of human information processing (e.g., Fitts and Posner 1967) distinguishes three basic processes: stimulus identification, response selection, and response programming. While stimulus identification and response programming are functions of stimulus and response properties, respectively, response selection is associated with the translation between stimuli and responses (Welford 1968).

Translation is the seat of the human “interface” between perception and action. Moreover, the effectiveness of translation processes at this interface is influenced to a large extent by the relation between perceptual inputs (e.g., stimuli) and motor outputs (e.g., responses). Since the seminal work of Fitts and colleagues (Fitts and Seeger 1953; Fitts and Deninger 1954), it has been repeatedly demonstrated that errors and choice reaction times to stimuli in a spatial array decrease when the stimuli are mapped onto responses in a spatially “compatible” manner. Fitts and Seeger (1953) referred to this finding as stimulus–response (S–R) compatibility and ascribed it to cognitive codes associated with the spatial locations of elements in the stimulus and response arrays. Presumably, it is the degree of coding and recoding required to map the locations of stimulus and response elements that determine the speed and accuracy of translation and thus response selection (e.g., Wallace 1971).

The relevance of studies of S–R compatibility to the domain of human factors engineering is paramount. It is now well understood that the design of an optimal HCI in which effective S–R translation facilitates fast and accurate responses is largely determined by the manner in which stimulus and response arrays are arranged and mapped onto each other (e.g., Bayerl, Millen, and Lewis 1988; Chapanis and Lindenbaum 1959; Proctor and Van Zandt 1994). As a user, we experience the recalibrating of perceptual-motor space when we take hold of the mouse and move it in a fairly random pattern when we interact with a computer for the first time. Presumably, what we are doing here is attempting to calibrate our actual movements to the resulting virtual movements of the cursor on the screen. Such recalibrations require neural networks and resources that are in addition to those typically activated during direct or standard mapping conditions (Snyder, Batista, and Andersen 1998). Thus, for optimal efficiency of functioning, it seems imperative that the system is designed to require as little recalibration as possible. Again, our contribution to the first edition of this handbook reviews our work on the area of S–R translation and the implications of this work for HCI (Chua, Weeks, and, Goodman 2003). We encourage those who are more interested in these issues to read that chapter. For the present chapter, we will instead outline some newer considerations and consequences for contexts in which there is a more direct translation between movements of the user and the effects of these actions in virtual space.

1.2 PERCEPTUAL-MOTOR INTERACTION: ATTENTION AND PERFORMANCE

The vast literature on selective attention and its role in the filtering of target from nontarget information (e.g., Cherry 1953; Treisman 1964a,b, 1986; Deutsch and Deutsch 1963; Treisman and Gelade 1980) has no doubt been informative in the resolution of issues in HCI pertaining to stimulus displays and inputs (e.g., the use of color and sound). However, attention should not be thought of as a unitary function, but rather as a set of information-processing activities that are important for perceptual, cognitive, and motor skills. Indeed, the evolution of HCI into the realm of augmented reality, teleoperation, gestural interfaces, and other areas that highlight the importance of dynamic perceptual-motor interactions, necessitates a greater consideration of the role of attention in the selection and execution of action. Recent developments in the study of how selective attention mediates perception and action and, in turn, how intended actions influences attentional processes, are poised to make just such a contribution to HCI. We will now turn to a review of these developments and some thoughts on their potential relevance to HCI.

1.2.1 Attention

We are all familiar with the concept of "attention" on a phenomenological basis. Even our parents, who likely never formally studied cognition, demonstrated their understanding of the essential characteristics of attention when they directed us to "pay attention" when we were daydreaming or otherwise not doing what was asked. They knew that humans, like computers, have a limited capacity to process information in that we can only receive, interpret, and act upon a fixed amount of information at any given moment. As such, they knew that any additional, nontask processing would disrupt the performance of our goal-task, be it homework, cleaning, or listening to their lecture. But what is "attention"? What does it mean to "pay attention"? What influences the direction of our attention? The answers to these questions are fundamental to understanding how we interact with our environment. Thus, it is paramount for those who are involved in the design of HCI to consider the characteristics of attention and its interactive relationship with action planning.

1.2.1.1 Characteristics of Attention

Attention is the collection of processes that allow us to dedicate our limited information-processing capacity to the purposeful (cognitive) manipulation of a subset of available information. Stated another way, attention is the process through which information enters into working memory and achieves the level of consciousness. There are three important characteristics of attention: (1) attention is selective and allows only a specific subset of information to enter the limited processing system; (2) the focus of attention can be shifted from one source of information to another; and (3) attention can be divided such that, within certain limitations, one may selectively attend to more than one source of information at a time. The well-known "cocktail party" phenomena (Cherry 1953) effectively demonstrates these characteristics.

Picture yourself at the last busy party or poster session you attended where there was any number of conversations continuing simultaneously. You know from your own experience that you are able to filter out other conversations and *selectively* attend to the single conversation in which you are primarily engaged. You also know that there are times when your attention is drawn to a secondary conversation that is continuing nearby. These *shifts* of attention can occur automatically, especially if you hear your name dropped in the second conversation, or voluntarily, especially when your primary conversation is boring. Finally, you know that you are able to *divide* your attention and follow both conversations simultaneously. However, although you are able to keep track of each discussion simultaneously, you will note that your understanding and contributions to your primary conversation diminish as you dedicate more and more of your attentional resources to the secondary conversation. The diminishing performance in your primary conversation is, of course, an indication that the desired amount of information processing has exceeded your limited capacity.

Although the "cocktail party" example outlined here uses auditory stimuli, the ability to select, divide, and shift attentional resources holds for different modalities (e.g., vision, proprioception) and across multiple modalities (e.g., one can shift from auditory stimuli to visual stimuli). Because vision is the dominant modality of information transfer in HCI, we will concentrate our discussion on visual selective attention. It should be noted, however, that there is a growing literature on cross-modal influences on attention, especially visual–auditory system interactions (e.g., Spence et al. 2000), that will be relevant in the near future. For those interested in a broader review of the characteristics of attention are encouraged to read our contribution to the second edition of the handbook (Welsh et al. 2007).

1.2.1.2 Shifts of Attention

Structural analyses of the retinal (photosensitive) surface of the eye have revealed two distinct receiving areas—the fovea and the perifoveal (a.k.a. peripheral) areas. The fovea is a relatively small area (about 2°–3° of visual angle) near the center of the retina, which has the highest concentration of color-sensitive cone cells. It is this high concentration of color-sensitive cells that provides the rich, detailed information that we typically use to identify objects. There are several important consequences of this structural and functional arrangement. First, because of the fovea's pivotal role in object identification and the importance of object identification for the planning of action and many other cognitive processes, visual attention is typically dedicated to the information received by the fovea. Second, because the fovea is such a small portion of the eye, we are unable to derive a detailed representation of the environment from a single fixation. As a result, it is necessary to constantly move information from objects in the environment onto the fovea by rapidly and accurately rotating the eye. These rapid eye movements are known as saccadic eye movements. Because of the tight link between the location of visual attention and saccadic eye movements, these rapid eye movements are referred to as overt shifts of attention.

Although visual attention is typically dedicated to foveal information, it must be remembered that the perifoveal retinal surface also contains color-sensitive cells and, as such, is able to provide details about objects. A *covert* shift of attention refers to any situation in which attention is being dedicated to a nonfoveated area of space. Covert shifts of attention are used when an individual wants or needs to maintain the fovea on a particular object while continuing to scan the remaining environment for other stimuli. Covert shifts of attention also occur immediately before the onset of an overt shift of attention or other type of action (e.g., Shepherd, Findlay, and Hockey 1986). For this reason, people are often able to identify stimuli at the location of covert attention before the acquisition of that location by foveal vision (i.e., overt attention) (Deubel and Schneider 1996).

Both overt and covert shifts of attention can be driven by stimuli in the environment or by the will of the performer. Shifts of attention that are driven by stimuli are known as *exogenous*, or bottom–up, shifts of attention. They are considered to be automatic in nature and thus, for the most part, are outside of cognitive influences. Exogenous shifts of attention are typically caused by a dynamic change in the environment such as the sudden, abrupt appearance (onset) or disappearance (offset) of a stimulus (e.g., Pratt and McAuliffe 2001), a change in the luminance or color of a stimulus (e.g., Folk, Remington, and Johnston 1992; Posner, Nissen, and Ogden 1978; Posner and Cohen 1984), or the abrupt onset of object motion (e.g., Abrams and Chirst 2003; Folk, Remington, and Wright 1994). The effects of exogenous shifts have a relatively rapid onset, but are fairly specific to the location of the dynamic change and are transient, typically reaching their peak influence around 100 ms after the onset of the stimulus (Cheal and Lyon 1991; Müller and Rabbitt 1989). From an evolutionary perspective, it could be suggested that these automatic shifts of attention developed because such dynamic changes would provide important survival information such as the sudden, unexpected appearance of a predator or prey. However, in more modern times, these types of stimuli can be used to quickly draw one's attention to the location of important information.

In contrast, performer-driven, or *endogenous*, shifts of attention are under complete voluntary control. The effects of endogenous shifts of attention take longer to develop, but can be sustained over a much longer period of time (Cheal and Lyon 1991; Müller and Rabbitt 1989). From an HCI perspective, there are advantages and disadvantages to the fact that shifts of attention can be under cognitive control. The main benefit of cognitive control is that shifts of attention can result from a wider variety of stimuli such as symbolic cues like arrows, numbers, or words. In this way, performers can be cued to locations or objects in the scene with more subtle or permanent information than the dynamic changes that are required for exogenous shifts. The main problem with endogenous shifts of attention is that the act of interpreting the cue requires a portion of the limited information-processing capacity and thus can interfere with, or be interfered by, concurrent cognitive activity (Jonides 1981).

Although it was originally believed that top–down processes could not influence exogenous shifts of attention (i.e., that dynamic changes reflexively capture attention regardless of intention), Folk, Remington, and Johnston (1992) demonstrated that this is not always the case. The task in the Folk et al. (1992) study was to identify a stimulus that was presented in one of four possible locations. For some participants, the target stimulus was a single abrupt onset stimulus (the target appeared in one location and nothing appeared in the other three locations), whereas for the remaining participants the target stimulus was a color singleton (a red stimulus that was presented at the same time as white stimuli that appeared in the other three possible locations). One-hundred and fifty milliseconds before the onset of the target, participants received cue information at one of the possible target locations. The cue information was either abrupt onset stimuli at a single location or color singleton information. Across a series of experiments, Folk et al. (1992) found that the cue tended to increase reaction times to the target stimulus when the cue information was presented at a location that was different from where the target subsequently appeared, indicating that attention had initially been exogenously drawn to the cue. Importantly, the cue stimuli only interfered with the identification of the target stimulus when the characteristics of cue stimuli matched the characteristics of the target stimulus (i.e., onset cue-onset target and color cue-color target conditions). When the characteristics of the cue did not match the target stimulus (i.e., onset cue-color target and color cue-onset target conditions), the location of the cue did not influence reaction times. Thus, these results reveal that dynamic changes only capture attention when the performer is searching for a dynamic change stimulus. Stated another way, it seems that "automatic" attentional capture is dependent on the expectations of the performer. Folk et al. suggested that people create an attention set in which they establish their expectations for the characteristics of the target stimulus. Stimuli meeting the established set will automatically capture attention, whereas stimuli that do not meet the established set will not.

Subsequent work on this contingent involuntary capture of attention effect has revealed that this attentional set can only be broadly-tuned in that it is most sensitive for discriminating between so-called static (e.g., color singletons) and dynamic (e.g., abrupt onset singletons) discontinuities. For example, Folk, Remington, and Wright (1994) found that a motion singleton (one object suddenly starting to move) and an offset singleton (one object suddenly disappearing) captured attention when participants were searching for an onset singleton target (see also Gibson and Kelsey 1998). Thus, when key press responses are required to a target that is characterized by a dynamic change in the environment, other dynamic change will fit the attentional set and capture attention. The obvious implication of these results is that the most efficient HCIs will be those for which the designer has considered perceptual expectations of the person controlling the system. As we will discuss in Section 1.2.1.3, however, consideration of the perceptual expectations alone is, at best, incomplete.

1.2.1.3 Action-Centered Attention

The majority of the literature reviewed thus far has involved experiments that investigated attentional processes through tasks that used simple or choice key press actions. Cognitive scientists typically use these arbitrary responses because (1) key press responses are relatively uncomplicated and provide simple measures of performance, namely reaction time and error; and (2) by using a simple response, the researcher assumes that they have isolated the perceptual and attentional processes of interest from additional complex motor programming and control processes. Although there are certainly numerous examples of HCI in which the desired response is an individual key press or series of key presses, there are perhaps as many situations in which more complicated movements are required. Indeed, mouse- and joystick-based interactions are in many ways complicated aiming movements. Further, as HCIs move increasingly into virtual reality, touchscreen, tangible interfaces, and other more complex environments, it will become increasingly important to consider the ways in which attention and motor processes interact. Thus, it will become more critical to determine if the same principles of attention apply when more involved motor responses are required. In addition, some cognitive scientists have suggested that, because human attention systems have developed through evolution to acquire the information required to plan and control complex actions, studying attention under such constrained response conditions may actually provide an incomplete or biased view of attention (Allport 1987, 1993). The tight link between attention and action is apparent when one recognizes that covert shifts of attention occur before saccadic eye movements (Deubel and Schneider 1996) and that overt shifts of attention are tightly coupled to manual aiming movements (Helsen et al. 1998, 2000). Such considerations, in combination with neuroanatomical studies revealing tight links between the attention and motor centers (Rizzolatti, Riggio, and Sheliga 1994), have led to the development of action-centered models of attention (Rizzolatti et al. 1987; Tipper, Howard, and Houghton 1999; Welsh and Elliott 2004a).

1.2.1.3.1 The Relationship between Attentional Capture and Action Coding

Recent research has demonstrated that the behavioral consequences of selecting and executing target-directed actions in the presence of action-relevant nontarget stimuli extend beyond the time taken to prepare and execute the movement (e.g., Meegan and Tipper 1998; Pratt and Abrams 1994). Investigations in our labs and others have revealed that the actual execution of the movement changes in the presence of distractors. For example, there are reports that movements will deviate toward (Welsh, Elliott, and Weeks 1999; Welsh and Elliott 2004a; Welsh et al. 2007; Song and Nakayama 2008; Carr, Phillips, and Meehan 2008; Buetti and Kerzel 2009) or away from (Howard and Tipper 1997; Tipper, Howard, and Jackson 1997; Welsh and Elliott 2004a,b) the nontarget stimulus. For a recent review of the effects of cognitive states on reaching movements, please see Song and Nakayama (2009).

Welsh and Elliott have developed the model of response activation to account for and integrate this research. Consistent with the conclusions of Tipper, Lortie, and Baylis (1992), Welsh and Elliott (2004a) based the model of response activation on the premise that attention and action processes are so tightly linked that the dedication of attention to a particular stimulus automatically initiates response-producing processes that are designed to interact with that stimulus. Responses are activated to attended stimuli regardless of the nature of attentional dedication (i.e., reflexive or voluntary). It is proposed that each time a performer approaches a known scene, a "response set" is established in working memory in which the performer identifies and maintains the characteristics of the expected target stimulus and the characteristics of the expected response to that stimulus. Thus, the response set in the model of response activation is an extension of the attentional set of Folk et al. (1992) in that the response set includes the performer's expectations of the target stimulus as well as preexcited (preprogrammed) and/or preinhibited response codes. Each stimulus that matches the physical characteristics established in the response set captures attention and, as a result, activates an independent response process. Stimuli that do not possess at least some of the expected characteristics do not capture attention and thus do not activate responses. Thus, if only one stimulus in the environment matches the response set, then that response process is completed unopposed and the movement emerges rapidly and in an uncontaminated form. However, under conditions in which more than one stimulus matches the response set, multiple response representations are triggered and subsequently race one another to surpass the threshold level of neural activation required to initiate a response. It is important to note that this is not a "winner-take-all" race where only the characteristics of the winning response influence the characteristics of actual movement alone. Instead, the characteristics of the observed movement are determined by the activation level of each of the competing responses at the moment of movement initiation. In this way, if more than one neural representation is active (or if one is active and one is inhibited) at response initiation, then the emerging response will have characteristics of both responses (or characteristics that are opposite to the inhibited response).

The final relevant element of the model is that the activation level of each response is determined by at least three interactive factors—the salience of the stimulus and associated response, an independent inhibitory process, and the time course of each independent process. The first factor, the salience or action-relevancy of the stimulus, is in fact the summation of a number of separate components including the degree attentional capture (based on the similarity between the actual and anticipated stimulus within the response set), the complexity of the response afforded by the stimulus, and the S–R compatibility. When attentional capture and S–R compatibility are maximized and response complexity is minimized, the salience of an individual response is maximized and the response to that stimulus is activated rapidly.

So, what implications does the model of response activation have for the design of HCI? In short, because the model of response activation provides a fairly comprehensive account of movement organization in complex environments, it could be used as the basis for the design of interfaces that consider the cognitive system as an interactive whole as opposed to separate units of attention and movement organization. One of the more obvious implications is that a designer should consider the time intervals between the presentation of each stimulus in a multiple-stimuli set, as this can have dramatic effects on the performer's ability to quickly respond to each stimulus (e.g., psychological refractory period—Telford 1931; Pashler 1994) and the physical characteristics of each response (Welsh and Elliott 2004a).

1.2.1.3.2 Spatial Coordinates of Attention in Different Action Contexts

Arguably the most influential work in the development of the action-centered models was the article by Tipper et al. (1992). Participants in these studies were presented with nine possible target locations, arranged in a three by three matrix, and were asked to identify the location of a target stimulus appearing at one of these locations while ignoring any nontarget stimuli presented at one of the remaining eight locations. The key innovation of this work was that Tipper and colleagues asked participants to complete a rapid aiming movement to the target location instead of identifying it with a key press. Previous studies of the reference frame of attention using key press responses had revealed that attention can work in retinotopic (e.g., Eriksen and Eriksen 1974), egocentric (e.g., Downing and Pinker 1985; Gawryszewski et al. 1987), and environmental (e.g., Hinton and Parsons 1988) coordinate systems. However, if there is a tight link between attention and action and the requirements of the action modulate, in part, the distribution of attention and attentional capture, then coordinate system used (and subsequent pattern of distractor interference effects observed) during aiming movements should be different from that used during key press responses. This difference in coordinate systems should be observed because the amount and type of information needed to successfully plan and complete aiming movements are different from that needed to successfully complete a key press response (see Section 1.1.2).

Consistent with traditional key press studies, Tipper, Lortie, and Baylis (1992) found that the presence of a distractor increased response times to the target. Although the finding of distractor interference in this selective reaching task was an important contribution to the field in and of itself, the key discovery was that the magnitude of the interference effects caused by a particular distractor location was dependent on the aiming movement being completed. Specifically, it was found that distractors (1) closer to the starting position of the hand (between the start position and the target) cause more interference than distractors farther from the starting position (the proximity-to-hand effect); and, (2) ipsilateral to the moving hand caused more interference than those in the contralateral side of space (the ipsilateral effect). Based on this pattern of interference, Tipper et al. (1992) concluded that attention and action are tightly linked such that the distribution of attention is dependent on the action that was being performed (i.e., attention was distributed in an action-centered coordinate system). Specifically, stimuli that afford actions that are more efficiently executed (i.e., movements of shorter amplitude [Fitts 1954] or into ipsilateral space [Fisk and Goodale 1985]) tend to capture attention to a greater degree (and cause more interference) than distractors that afford less-efficient responses (i.e., movements of longer amplitude or into contralateral space; see also, Tipper, Meegan, and Howard 2002).

Although the study of Tipper et al. (1992) provided critical initial insights into the issue of response efficiency and the action-dependent patterns of interference, additional research has revealed that this pattern of interference is modulated by the characteristics of the environment and the task. For instance, Keulen et al. (2002) have demonstrated that the distance between targets and distractors in the environment alters the attentional frame of reference used during reaching movements. In support of Tipper et al. (1992)

action-centered frame of reference, they found that distractors closer to the start position of the hand caused more interference than distractors beyond the path of the reaching movement (i.e., a proximity-to-hand effect) when there was a large distance (20 mm) between the target and distractor locations. In contrast, when the target and distractor locations were close (5 mm) to each other, a symmetrical pattern of interference was observed in which distractors on either side of the target caused the same amount of interference (i.e., no proximity-to-hand effect was observed). The authors suggested that this shift in the pattern of interference occurred because the planning and control stages of aiming movements require different frames of reference (action-centered and environmental, respectively). These data support the action-centered view in that the patterns interference was even dependent on the stage of action planning and execution. Within the realm of HCI, these data highlight the need for careful consideration of the spatial arrangement of stimuli in the environment because even small changes in the array can alter the efficiency of target engagement.

1.2.1.3.3 The Capture of Attention in Different Action Contexts

As reviewed in Section 1.2.1.3.2, initial investigations into action-centered attention were focused primarily on the influence that the spatial location of distractors with respect to the target had on the planning and execution of action (e.g., Meegan and Tipper 1998; Lyons et al. 1999; Pratt and Abrams 1994; Tipper et al. 1992). In that context, an action-centered framework has offered a useful perspective for the spatial organization of perceptual information presented in an HCI context. However, the reason for engaging a target in an HCI task is because the target symbolically represents an outcome or operation to be achieved. Indeed, this is what defines an icon as a target—target features symbolically carry a meaning that defines it as the appropriate target. Whether by intuition and trial and error, or through consideration of the research on attentional capture (e.g., Folk et al. 1992), programmers have already used a variety of dynamic changes to the stimulus characteristics (e.g., suddenly appearing, blinking, moving, growing, etc.) to draw our attention to certain objects and in the hopes of facilitating target engagement. Although there is little doubt that the dynamic stimuli are, in large part, successful in achieving these goals, recent investigations of how the context of the response influence perception and attention suggest that target engagement may be made more efficient through consideration of the response mode, the requirements of the actions system, and the relationship between the stimulus and the desired response.

As an initial illustration of the tight link between perceptual-motor processes, there is a growing body of evidence revealing how the characteristics of the prepared action influence the processing of certain visual stimuli. For instance, Lindemann and Bekkering (2009; see also Craighero et al. 1999) have shown that the degree of congruency between the action goal and the characteristics of an irrelevant stimulus can facilitate reaction times to initiate the movement. Participants in the study were told to reach out and grasp an X-shaped object as if they were going to turn it clockwise or counterclockwise. They were told in advance which type of movement they would be making and to wait for a "go" signal before initiating the movement that they had prepared. The "go" signal was apparent motion of an object in either a clockwise or counterclockwise direction. It was found that the participants initiated their movements more rapidly when the apparent motion of the "go" signal was congruent with the movement that they had prepared (e.g., a clockwise movement with a clockwise rotating stimulus) than when the apparent motion was incongruent with the prepared movement (e.g., a clockwise movement with a clockwise rotating stimulus). This congruency effect is consistent with other research and demonstrates that prepared movements enhance the perception of characteristics of objects that are related to the to-be-performed movement. For example, the preparation of grasping movements enhances the detection of targets that varied by size, whereas the preparation of pointing movements enhances the detection of targets that varied by luminance (Wykowska, Schubo, and Hommel 2009; see also, Symes et al. 2008).

While the research described in the previous paragraph suggests that perception of specific features is enhanced in an action-specific manner, recent work from our lab suggests that attentional capture is likewise modified by the requirement of the motor system. Specifically, Welsh and Pratt (2008) found that the attentional capture by some dynamic changes is different when key press and aiming responses are required. In this study, participants were asked to identify the location of an onset or offset target stimulus while ignoring a distractor stimulus of the opposite characteristics (i.e., onset targets were paired with offset distractors and vice versa). In separate experiments, participants responded to the target stimulus with a choice key press response or an aiming movement to the target location. Consistent with the findings of Folk et al. (1992) and Folk et al. (1994), interference effects were observed when an offset distractor was presented with an onset target and when an onset distractor was paired with an offset target. When aiming responses were required, however, inference effects were only observed when an onset distractor was presented with an offset target. The offset distractor did not cause an interference effect when participants were aiming to an onset target. Stated another way, the results indicated that an onset distractor slowed responding to an offset target in both key press and aiming tasks. An offset distractor, however, only interfered with task performance when a key press was required.

It was proposed that this action-dependent pattern of interference effects emerged because the action system modified the attentional set, thereby influencing what stimulus features capture attention and those that do not, based on the salience of the stimulus feature for the requirements of the to-be-performed action. Because key press tasks are ballistic in nature, a constant source of stable visual information is not needed to ensure accurate completion. As a result, any dynamic discontinuity is as salient as any other and can capture attention. In contrast, because the accuracy of aiming movements depends on a continual source of stimulus information for feedback-based control, offset and onset stimuli represent the two extreme ends of saliency to the motor system (with onsets at the maximally salient end and offset at the minimally salient end). As a result, offset stimuli have a very low salience to the motor system and are very unlikely to capture attention when an aiming response is required. In contrast, because onset stimuli are highly salient to the motor system, they are very likely to capture attention when aiming responses are required, regardless of the features of the target stimulus. Thus, it seems that the context of the action and the requirements of the motor system to ensure the accurate completion of the response help to shape the attentional set and what does and does not capture attention (see also Higgins and Welsh, submitted; Welsh and Zbinden 2009).

Similar action-specific interference effects to those observed in our lab have been shown across pointing and grasping actions (Bekkering and Neggers 2002; Weir et al. 2003), pointing and verbal responses (Meegan and Tipper 1999), and different types of pointing responses (Meegan and Tipper 1999; Tipper, Meegan, and Howard 2002). In sum, there is growing evidence that traditional conception of the information-processing stream as serial series of events with action only occurring after perception and cognition stages are completed is in need of revision. The research reviewed here suggests that the action system has what would traditionally be considered as an "upstream" effect and plays an important role in shaping perception and attention. From an applied perspective, now that HCI is moving into virtual reality and other types of assisted response devices, it will become increasingly important to consider the required and/or anticipated action when designing HCI environments. Specifically, this work on the spatial layout (e.g., Keulen et al. 2002) and the characteristics of the stimuli (e.g., Welsh and Pratt 2008) highlights the need for the designer to consider the interactions among perception, attention, and motor processing because there are some situations in which the transfer from simple to complex movements is not always straightforward.

1.2.1.4 Summary

Taken into the realm of HCI, it is our position that the interplay between shifts of attention, spatial compatibility, and object recognition will be a central human performance factor as technological developments continue to enhance the "directness" of direct-manipulation systems (cf. Shneiderman 1983, 1992). Specifically, as interactive environments become better abstractions of reality with greater transparency (Rutkowski 1982), the potential influence of these features of human information processing will likely increase. Thus, it is somewhat ironic that the view toward virtual reality, as the solution to the problem of creating the optimal display representation, may bring with it an "unintended consequence" (Tenner 1996). Indeed, the operator in such an HCI environment will be subject to the same constraints that are present in everyday life.

The primary goal of human factors research is to guide technological design in order to optimize perceptual-motor interactions between human operators and the systems they use within the constraints of maximizing efficiency and minimizing errors. Thus, the design of machines, tools, interfaces, and other sorts of devices utilizes knowledge about the characteristics, capabilities, as well as limitations, of the human perceptual-motor system. In computing, the development of input devices such as the mouse and graphical user interfaces was intended to improve human–computer interaction. As technology has continued to advance, the relatively simple mouse and graphical displays have begun to give way to exploration of complex gestural interfaces and virtual environments. This development may perhaps, in part, be a desire to move beyond the "artificial" nature of such devices as the mouse, to ones that provide a better mimic of reality. Why move an arrow on a monitor using a hand-held device to point to a displayed object, when instead, you can "reach" and "interact" with the object? Perhaps such an interface would provide a closer reflection of real-world interactions—and the seeming ease with which we interact with our environments, but also subject to the constraints of the human system. With this in mind, we now turn to an alternative approach to perceptual-motor interactions that we believe may point us in some exciting new directions.

1.3 COMMON CODING ACCOUNTS OF PERCEPTUAL-MOTOR INTERACTIONS

At the same time that the research on action-centered attention and perception is gaining momentum, a new approach to cognition has begun to emerge broadly termed "embodied cognition." This approach argues that, among other things, there is a bidirectional relationship between the body and cognition such that actions are influenced by cognitive operations and cognitive operations are influenced by movements and the body's action state. In many ways, cognition is considered to be a form of action. One of the key mechanisms that is considered to support this two-way connection between the body and cognition is a common representation in the brain that codes both the action plan and the sensory consequences of the action plan (the effects the action will have on the environment). It is this specific common coding mechanism that differentiates this theory from the modified views of the traditional information-processing theories reviewed in Section 1.2.1.3. On a functional level, it is suggested that these common codes connect the perception, execution, and imagination of movements and, as a result, can also help to shape other cognitive processes. Although there is a literature on the connections between action and a variety of cognitive processes, we will focus here on the relevant literature related to the interactions among action, perception, and imagination.

The origins of the common coding approach can be found in the seminal text of William James (1890). The more modern and in-depth development of this idea was first articulated by Prinz (1992) and has been refined and expanded as data accumulate (see, Decety 2002; Hommel et al. 2001; Prinz 2005). Simply put, a central outcome of this common coding mechanism is that perception and action are intimately linked such that the activation of one component automatically activates the coupled component. The planning of an action automatically activates a representation of the sensory consequences of the action and, conversely, perception or imagination of an effect automatically activates a representation of the action(s) that can bring about that effect. As a result, one can activate or simulate an action by conceiving of the desired effects on the environment and the effects of a planned action on the environment can be anticipated with the activation or simulation of the response.

A suggested consequence of this coding is that the motor system is activated when humans perceive and imagine movement-related information. This motor system activation and connection between movement (activation of motor representations), observation of movements (activation of perceptual representations), and imagination of movements (covert activation of motor and perceptual representations), then leads to the preferences and biases of our own movements, which can guide the way we perceive and imagine other movements and actions, and may also influence the way we process representations that embed movements (such as verbs). Consistent with these ideas, recent work has extended this effect to language and concept processing, showing that there is motor activation while imagining words encoding movements, and processing sentences involving movements (Bergen, Chang, and Narayan 2004; Wilson and Gibbs 2007; Holt and Beilock 2006; Barsalou 1999).

A common instance of the embodied resonance and simulation process that may involve the common codes is familiar to cinema goers: while watching an actor moving along a precipice, viewers may move their arms and legs or displace body weight to one side or another, based on what they would like to see the actor doing in the scene. Similar effects are seen in sports fans watching athletes perform and novice video game players interacting with their virtual character. Such "simulation" of others' actions may also underlie our ability to project ourselves into different character roles as well. For instance, this effect may explain why we are emotionally moved by a dramatic film scene: we simulate the characters' movements and emotional expressions using our own body and, as a result, recreate their emotional states.

In implementation terms, common coding can be thought of as an artificial neural network encoding both action and perception elements, where the activation of one type of element automatically activates the other elements (associative priming), similar to connectionist implementations of semantic priming (Cree, McRae, and McNorgan 1999). Imagination of movement, in this view, would be a form of implicit activation of the action network. Recent modeling work has shown how such common coding could arise purely through agent–environment interactions, when agents move from not using any representations (being purely reactive) to a strategy of using stored structures in the world/head. In addition, this model shows that common coding can arise from both evolutionary and within-lifetime learning (Chandrasekharan and Stewart 2007).

Most of the evidence for common coding is derived from behavioral studies in which it is assessed how actions in one medium (e.g., imagination) leads to a difference in reaction time or accuracy in another medium (e.g., execution). The following is a brief review of the experimental evidence for different types of interactions. For the sake of space and relevance to HCI, our review will focus on this behavioral evidence for common coding. It should be noted, however, that this behavioral evidence is supported by neurophysiological experiments, including imaging, transcranial magnetic stimulation (TMS), and patient studies (for a comprehensive review, see Rizzolatti and Craighero 2004; or Brass and Heyes 2005). Finally, for the sake of brevity, we will focus our discussion on the less intuitive and more relevant research on the implications of the common coding system for perception–action and imagination–action relationships. We focus on these relationships because we have used this work as the theoretical basis for a collaborative project to develop a novel tangible interface that we will highlight at the end of the chapter.

1.3.1 Perception–Action Common Coding

If common coding holds and the perception of an action automatically activates the observed action codes in the observer, then two distinct predictions can be made. The first prediction is that the observation and perception of a movement should negatively influence the concurrent performance of a movement when the observed and executed actions are incompatible because different action codes are activated in the motor system through execution and observation. Thus, the codes of observed action should interfere with the codes of the action that is to be executed. This interference effect would be similar to the trajectory deviation effects caused by competing response codes observed in the action-centered attention studies reviewed earlier in the chapter (e.g., Welsh and Elliott 2004a).

In support of the common coding hypothesis, Kilner, Paulignan, and Blakemore (2003; see also Brass, Bekkering, and Prinz 2001) found that there was more variability in the performance of a rhythmic movement pattern when participants observed another individual performing an incompatible versus a compatible rhythmic pattern. Specifically, when participants were performing a rhythmic up-and-down movement pattern with their arms, there was more horizontal deviation in movement pattern when they observed another person performing a horizontal movement pattern than when the observed person performed a vertical movement pattern. Critically, this interference effect did not occur when the participants observed similar compatible and incompatible movement patterns being executed by a robot arm. This contrast in effects of the human and the robot suggests that the activation of the common codes through observation may be sensitive to the characteristics of the observed motion and/or the intentionality the observer is able to attribute to the observed actor.

The second, and probably more relevant, prediction is that the perception of actions should be affected by performance of those actions because recent or extensive execution improves the coding of and familiarity with the perceptual consequences of the action. There are a number of lines of evidence that are consistent with this prediction. One line of evidence is the repeated finding that people are better able to recognize actions after having practiced the action patterns. For example, Casile and Giese (2006) found that people were better able to visually recognize a specific movement pattern faster than other movement sequences after learning the movement pattern. Critically, because participants were blindfolded during the learning of the task, the improvement in *visual* recognition was based on verbal and haptic feedback alone. In a related set of studies, Knoblich, Sebanz, and colleagues (see Knoblich and Sebanz 2006 for a review) have shown that people can accurately identify their own action patterns from those of other people. Presumably, people are very accurate at recognizing their own actions because they have a lifetime of experiencing and building of knowledge of their own action–effect relationships.

In addition to the work on recognition, this effect of learned actions seems to extend to preference judgments. When skilled and novice typists were asked to pick between dyads of letters (such as FV and FJ), the skilled typists preferred dyads that would be typed with less interference (i.e., different fingers), whereas novices showed no preference. Moreover, a motor task performed in parallel to the dyad preference judgments lowered skilled typists' preference, but only when the motor task involved the specific fingers that would be used to type the dyads (Beilock and Holt 2007). This preference effect has been generalized recently by Topolinski and Strack (2009), who showed that the mere exposure effect (MEE; stimuli that are repeatedly encountered are increasingly liked) is dependent on motor simulations. They showed that chewing gum while evaluating stimuli destroyed MEEs for words, but not for visual characters. However, kneading a ball with the hand left both MEEs unaffected. They argued that this effect stems from individuals representing stimuli by covertly simulating the sensorimotor processes that run when the stimuli are perceived or acted on. Chewing disrupts this process, kneading does not. These preference effects have recently been used to explain the strong identification players develop with video game characters (Chandrasekharan et al. 2010).

1.3.2 Imagination–Action Common Coding

We believe the most straightforward and convincing demonstration of the involvement of the motor system in imagination, at least in the imagination of actions, is the repeated finding that the time to mentally execute actions closely corresponds to the time it takes to actually perform them (Decety 2002; Jeannerod 2006; Young, Pratt, and Chau 2009). However, it has also been shown that responses beyond voluntary control (such as heart and respiratory rates) are activated by imagining actions to an extent proportional to that of actually performing the action (Decety 2002). In sum, these data suggest that imagination of these actions involves the activation of response codes, with these response codes running offline and generating many of the same physiological effects that would be generated during execution, although to a diminished degree. The connections between the motor system and imagination extend beyond the simulation of motor tasks to other cognitive activities (e.g., Hegarty 2004; Martin and Schwartz 2005; Nersessian 2002, 2008). We will center our discussion, however, on mental rotation.

The main prediction of this work is that, if cognitive processes such as imagination and mental rotation engage the common coding system, then these cognitive processes should be affected by concurrent action execution and vice versa. To test the prediction that action planning and execution influences cognition, Wohlschlager (2001; see also Wexler, Kosslyn, and Berthoz 1998) asked participants to mentally rotate an object while they were planning an action or actually moving their hands or feet in a direction that was compatible or incompatible with the direction of the mental rotation. Consistent with predictions based on the notion of common coding, performance on the mental rotation suffered when the direction of action was incompatible with the direction of mental rotation and performance improved when the direction of action was compatible with the mental rotation.

Although the involvement of our action system in cognition may facilitate efficient processing, the limitations of our motor system may likewise limit or hinder cognitive functioning. For example, it has recently been shown that people with writer's cramp (a focal hand dystonia characterized by constant contractions of the muscles of the hand and forearm that limit hand use) take more time to complete certain mental rotation tasks than their peers without neurological disorders. Interestingly, the difficulties in mental rotation seem to be specific to images of the affected limb (i.e., rotating pictures of hands). The time it took people with focal hand dystonia to rotate pictures of nonbody parts (e.g., houses and cars) were not different from their peers without dystonia (Fiorio, Tinazzi, and Agiloti 2006). Likewise, Kosslyn (1994) reports that participants need more time to perform mental rotations that are physically awkward. These data suggest that common coding may restrict or limit our ability to imagine novel actions and movements. Thus, although our action system may be engaged to facilitate certain cognitive processes, its role is limited by our action repertoire.

1.3.3 Summary

Through this review, we have attempted to concisely summarize the critical features of common coding theories and the evidence that supports these views. Although this area is, in many ways, in its infancy, there is a clear growing body of evidence supporting a common code system linking execution, perception, and imagination of movement and that this system can be accessed to support a wide variety of cognitive processes. The vast majority of the research in this area has been directed to testing and expanding the theoretical aspects of common coding. We believe, however, that there is tremendous potential for the principles outlined in common coding theory to shape and enhance HCI. In fact, this theoretical approach has recently been used to derive novel embodied interaction designs. We will describe this development and some potential applications in the second half of the following section.

1.4 PERCEPTUAL-MOTOR INTERACTION IN APPLIED TASKS: A FEW EXAMPLES

As we mentioned at the outset of this chapter, the evolution of computers and computer-related technology has brought us to the point at which the manner with which we interact with such systems has become a research area in itself. Current research in motor behavior and experimental psychology pertaining to attention, perception, action, and spatial cognition is poised to make significant contributions to the area of HCI. In addition to the continued development of a knowledge base of fundamental information pertaining to the perceptual-motor capabilities of the human user, these contributions will include new theoretical and analytical frameworks that can guide the study of HCI in various settings. In this final section, we highlight just a few specific examples of HCI situations that offer a potential arena for the application of the basic research that we have outlined in this chapter.

1.4.1 Attention Cueing for Military Target Detection

Combat identification of friends and enemies is essential for mission effectiveness and the prevention of friendly fire. The software that projects images to the operator's displays, including images from unmanned aerial vehicles and on head mounted displays (HMDs), can cue attention to possible target locations. In each situation, the user is required to navigate and engage targets in the real world, while attempting to perform a detection and identification on the interactive display. The use of this assistive software creates a dual task in which the operator must divide his or her attention between the separate tasks in order to complete the job successfully. Although the identification cues can provide great opportunities to facilitate the detection of critical information, they could also decrease performance by creating distracting clutter on the display (Yeh and Wickens 2001a,b). These effects are magnified as the cue reliability decreases and, in the case where the task can be performed easily without the cue, it has been shown that imperfect cues may hinder performance (Maltz and Shinar 2003). In this context, an error of commission (i.e., the cue indicates a nontarget) has much greater behavioral consequences than an error of omissions (i.e., the technology fails to cue a target) (Maltz and Shinar 2003). Thus, it is imperative that the cue stimuli involved in the secondary identification task be carefully designed to ensure the efficient processing of this information to allow as much of the attentional resources as possible to be available for the real-world tasks of target engagement.

An HMD can assist with target detection because it overlays critical cue information over the actual environment, reducing the scanning time required to sample and attend both the display and the environment. An HMD also allows for cueing in *x* and *y* coordinates and the use of *conformal imagery* in which cues or information is presented in a world-referenced frame rather than a screen-referenced frame eliminating the need for the user to transfer between reference frames (Yeh, Wickens and Seagull 1999). Users are also better at recovering from cueing errors using an HMD (Yeh et al. 2003). However, HMDs are especially susceptible to the detrimental effects of clutter as the user is expected to attend concurrently to information both on the display and in the environment.

The majority of studies find that cueing assists the user in detecting the target more quickly and accurately (e.g., Maltz and Shinar 2003); however, there is often a cost for detecting uncued targets (e.g., Yeh et al. 2003) and an increase in false alarms (Yeh and Wickens 2001a,b). The cost may result from "attentional tunnelling" where the participants fail to direct their attention to areas outside of the cue. The tunnelling may result from the user creating an attention set (Folk et al. 1992; Folk et al. 1994) for specific cue features. This attentional set may increase the chances of these salient cue stimuli capturing attention, but at the same time reduce the chances that stimuli not in the set (i.e., uncued targets) capture attention. Overall, cueing can assist the user in directing attention in difficult detection tasks as long as the cue is sufficiently reliable and does not induce clutter into the visual scene.

1.4.2 Deriving Novel Interaction Designs from Common Coding

Although there are clear implications for the research outlined above for the design of stimuli in virtual environments, we also believe that the principles of perceptual-cognitive-motor interactions outlined above should shape and enhance the interface devices that are used to translate our action goals into virtual environments. In fact, two of us (Timothy N. Welsh and Sanjay Chandrasekharan) have been involved in the recent development of a novel interaction device (see Mazalek et al. 2010). The goal of the research was to develop a device that more effectively mapped the actions of the user to the movements of the avatar. The rationale for this goal being that, by translating the user's own actions onto the avatar, there will be a shorter recalibration period, and the user can more easily relate to the avatar and respond more efficiently in the virtual environment. An additional, yet to be tested, potential consequence of this more direct relationship between user and avatar, is that once the user has identified with the avatar's movements, it is possible that the user can then learn from the avatar if it moves in a novel action pattern that is physically consistent with the movements of the user. Thus, a more direct translation from user to avatar might not only facilitate performance in virtual environments, but it might also assist in the learning and development of the user.

The development of this device is rooted in the common coding theory (e.g., Prinz 1997). A particularly informative set of findings are that individuals recognize their own actions more accurately than the action patterns of other people, even when all that is available is a very information-poor rendition such as point-light displays (see Knoblich and Sebanz 2006, for a review). This own-action advantage is thought to arise because the observer's motor system is involved in the perception process. Because the motor system of the observer is trained to their own actions and the sensory consequences of those actions, it is thought that viewing their own actions more efficiently activates their motor system and the tightly linked common perceptual codes. This more efficient activation of the common codes then allows the individuals to identify their own actions more accurately than those of other people. Extending these findings to HCI, we reasoned that a user would identify more closely with a virtual character in a virtual environment if that virtual character encodes the player's own actions as opposed to movement primitives common to all, or at least a subset of, characters.

Based on this experimental and theoretical work, a control interface was developed to more directly map the user's own actions onto a virtual character in a real-time virtual environment. The device that was developed was a wearable, jointed puppet whose limbs are attached to the limbs of the user so that the limbs of the puppet move along with the hands, legs, and neck of the user. Potentiometers are located at the joints so that the changes in joint angle of the puppet can be transferred to a virtual character (Mazalek et al. 2010). As an initial testing of the puppet system, we recently examined if the same "own-action" advantage (people can recognize their own movements better than others, Knoblich and Sebanz 2006) was present when their movements were represented by a virtual character. Consistent with previous work, we have found that individuals were able to identify abstract representations (point-light displays) of their own actions when the representations were created by affixing small lights to actor's actual body (see Mazalek et al. 2009) and when a player's movements are transferred to an avatar using the puppet (Mazalek et al. 2010). The advantages persist even when the point-light walkers were presented in altered body sizes (Mazalek et al. 2009). Thus, we feel confident that the movements of an individual can be effectively transferred to virtual characters through the puppet device and that people may be able to identify with (embody) these characters when this transfer of movement patterns is successful.

Although our initial development and testing of the device seems positive, the interface continues to evolve. As the interface improves, our view to the possible applications of this system, beyond real-time interaction, expands. For example, we are opening a second line of research in which we are trying to exploit the link between action, cognition, and imagination. Extending the results from the research reviewed above and the theoretical relationship between action, cognition, and imagination, we hypothesized that novel movements executed by the embodied avatar may improve imagination of novel movements, thus improving players' ability to execute creative cognitive processes such as mental rotation. To facilitate this learning effect, however, the "embodied" virtual characters (characters encoding the player's own actions) would need to execute movements on screen that are impossible for the actual user to perform. Further, the user will lose some control of the embodied avatar when the avatar executes novel movements, such as back-flips (as this would require the user also doing back-flips). Thus, the puppet-controlled avatar will need to retain the movement patterns of the user while executing these physically impossible movements. This is an interesting application challenge, where we need to maintain a fine line between control and no-control, with self-recognition elements of the former situation retained/continued into the latter situation.

As an initial attempt to solve this issue, we have developed a game in which the cameras around the avatar rotate slowly, giving the impression of the avatar rotating in space. Objects then appear close to the avatar, and the user's task is to touch these objects using the puppet interface. Our preliminary results reveal that playing this game using the puppet leads to improved performance on the game and a mental rotation task compared with playing the game using standard game interfaces, such as keyboards and game controllers. These and other experimental applications are still under development. While we are hopeful that the puppet device will achieve all our aims, we feel that, regardless of the outcome, this entire line of research is a powerful example of how theoretical considerations of perceptual-cognitive-motor interactions can be used to inform HCI development and, likewise, this technological development can lead to new methods for testing and enhancing the theory on which the technology was based. For a wider discussion of how common coding theory can help in deriving novel interaction modes, see Chandrasekharan et al. (2010).

1.5 SUMMARY

The field of HCI offers a rich environment for the study of perceptual-motor interactions. The design of effective human–computer interfaces has been, and continues to be, a significant challenge that demands an appreciation of the entire human perceptual-motor system. The information-processing approach has provided a dominant theoretical and empirical framework for the study of perceptual-motor behavior in general, and for consideration of issues in HCI and human factors in particular. Texts in the area of human factors and HCI (including the present volume) are united in their inclusion of chapters or sections that pertain to the topic of human information processing. Moreover, the design of effective interfaces reflects our knowledge of the perceptual (e.g., visual displays, use of sound, graphics), cognitive (e.g., conceptual models, desktop metaphors), and motoric constraints (e.g., physical design of input devices, ergonomic keyboards) of the human perceptual-motor system.

Technological advances have undoubtedly served to improve the HCI experience. For example, we have progressed beyond the use of computer punch cards and command-line interfaces to more complex tools such as graphical user interfaces, speech recognition, and tangible control systems. As HCI has become not only more effective, but by the same token more elaborate, the importance of the interaction between the various perceptual, cognitive, and motor constraints of the human system has come to the forefront. In our previous chapters, we presented overviews of some topics of research in action-centered attention and in S–R compatibility in perceptual-motor interactions that we believed were relevant to HCI. In the present chapter, we have added an overview of common coding theories of cognition. We believe that the relevance of the research and theoretical considerations discussed in this chapter for HCI cannot be underestimated. Clearly, considerable research will be necessary to evaluate the applicability of both of these potentially relevant lines of investigation to specific HCI design problems. Nevertheless, the experimental work to date leads us to conclude that the motor system is not simply responsible for outputting the results of perceptual and cognitive processing, but in fact has a critical and active role in shaping perception and cognition. For this reason, an effective interface must be sensitive to the perceptual and action expectations of the user, the specific action associated with a particular response location, the action relationship between that response and those around it, and the degree of translation required to map the perceptual-motor workspaces.

ACKNOWLEDGMENTS

We would like to recognize the financial support of the Natural Sciences and Engineering Research Council of Canada, Alberta Ingenuity Fund, and the National Science Foundation.

REFERENCES

Abrams, R. A., and S. E. Chirst. 2003. Motion onset captures attention. *Psychol Sci* 14:427–32.

Adam, J. J., R. Mol, J. Pratt, and M. H. Fischer. 2006. Moving farther but faster: An exception to Fitts's law. *Psychol Sci* 17:794–8.

Adams, J. A. 1971. A closed-loop theory of motor learning. *J Mot Behav* 3:111–50.

Adams, J. A., and Dijkstra, S. 1966. Short-term memory for motor responses. *J Exp Psychol* 71:314–8.

Allport, A. 1987. Selection for action: Some behavioural and neurophysiological considerations of attention and action. In *Perspectives on Perception and Action*, ed. H. Heuer, and A. F. Sanders. 395–419. Hillsdale, NJ: Erlbaum.

Allport, A. 1993. Attention and control: Have we been asking the wrong questions? A critical review of twenty-five years. In *Attention and Performance 14: Synergies in Experimental Psychology, Artificial Intelligence, and Cognitive Neuroscience*, ed. D. E. Meyer and S. Kornblum, 183–218. Cambridge, MA: MIT Press.

Barsalou, L. W. 1999. Perceptual symbol systems. *Behav Brain Sci* 22:577–660.

Bayerl, J., D. Millen, and S. Lewis. 1988. Consistent layout of function keys and screen labels speeds user responses. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, 334–46. Santa Monica, CA: Human Factors Society.

Beilock, S. L., and L. E. Holt. 2007. Embodied preference judgments: Can likeability be driven by the motor system? *Psychol Sci* 18:51–7.

Bekkering, H., and S. F. W. Neggers. 2002. Visual search is modulated by action intentions. *Psychol Sci* 13:370–4.

Bergen, B., N. Chang, and S. Narayan. 2004. Simulated action in an embodied construction grammar. In *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*, ed. K. D. Forbus, D. Gentner, and T. Regier, 108–13. Hillsdale, NJ: Lawrence Erlbaum.

Brass, M., H. Bekkering, and W. Prinz. 2002. Movement observation affects movement execution in a simple response task. *Acta Psychol* 106:3–22.

Brass, M., and C. Heyes. 2005. Imitation: Is cognitive neuroscience solving the correspondence problem? *Trends Cogn Sci* 9:489–95.

Broadbent, D. E. 1958. *Perception and Communication*. New York, NY: Pergamon.

Brooks, V. B. 1974. Some examples of programmed limb movements. *Brain Res* 71:299–308.

Buetti, S., and D. Kerzel. 2009. Conflicts during response selection affect response programming: Reactions toward the source of stimulation. *J Exp Psychol Hum Percept Perform* 35:816–34.

Card, S. K., W. K. English, and B. J. Burr. 1978. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* 21:601–13.

Carr, S. M., J. G. Phillips, and J. W. Meehan. 2008. Non-target flanker effects on movement in a virtual action centered reference frame. *Exp Brain Res* 184:95–103.

Casile, A., and M. A. Giese. 2006. Non-visual motor learning influences the recognition of biological motion. *Curr Biol* 16:69–74.

Chandrasekharan, S., A. Mazalek, Y. Chen, M. Nitsche, and A. Ranjan. 2010. Ideomotor Design: using common coding theory to derive novel video game interactions. *Pragmat Cogn* 18:313–39.

Chandrasekharan, S., and T. C. Stewart. 2007. The origin of epistemic structures and proto-representations. *Adapt Behav* 15:329–53.

Chapanis, A., and L. E. Lindenbaum. 1959. A reaction time study of four control-display linkages. *Hum Factors* 1:1–7.

Cheal, M. L., and D. R. Lyon. 1991. Central and peripheral precuing of forced-choice discrimination. *Q J Exp Psychol* 43A:859–80.

Cherry, E. C. 1953. Some experiments on the recognition of speech, with one and with two ears. *J Acoust Soc Am* 25:975–9.

Chua, R., and D. Elliott. 1993. Visual regulation of manual aiming. *Hum Mov Sci* 12:365–401.

Chua, R., D. J. Weeks, and D. Goodman. 2003. Perceptual-Motor Interaction: Some Implications for HCI. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emgering Applications*, ed. J. A. Jacko, and A. Sears, 23–34. Hillsdale, NJ: Lawrence Earlbaum Assoc.

Craighero, L., L. Fadiga, G. Rizzolatti, and C. Umiltà. 1999. Action for perception: A motor-visual attentional effect. *J Exp Psychol Hum Percept Perform* 25:1673–92.

Cree, G. S., K. McRae, and C. McNorgan. 1999. An attractor model of lexical conceptual processing: Simulating semantic priming. *Cogn Sci* 23:371–414.

Crossman, E. R. F. W., and P. J. Goodeve. 1983. Feedback control of hand movement and Fitts' Law. *Paper presented at the meeting of the Experimental Psychology Society*, Oxford, July 1963. Published in *Q J Exp Psychol* 35A:251–78.

Decety, J. 2002. Is there such a thing as a functional equivalence between imagined, observed and executed actions. In *The Imitative Mind: Development, Evolution and Brain Bases*, ed. A. N. Meltzoff and W. Prinz, 291–310. Cambridge: Cambridge University Press.

Deubel, H., and W. X. Schneider. 1996. Saccade target selection and object recognition: Evidence for a common attentional mechanism. *Vis Res* 36:1827–37.

Deutsch, J. A., and D. Deutsch. 1963. Attention: Some theoretical considerations. *Psychol Rev* 70:80–90.

Downing, C. J. and S. Pinker. 1985. The spatial structure of visual attention. In *Attention and Performance XI*, ed. M. I. Posner and O. S. M. Marin, 171–87. Hillsdale, NJ: Erlbaum.

Elliott, D., and M. A. Khan. 2010. *Vision and Goal-directed Movements: Neurobehavioral Perspectives*. Champaign, LI: Human Kinetics.

Elliott, D., R. G. Carson, D. Goodman, and R. Chua. 1991. Discrete vs. continuous visual control of manual aiming. *Hum Mov Sci* 10:393–418.

English, W. K., D. C. Engelhart, and M. L. Berman. 1967. Display-selection techniques for text manipulation. *IEEE Trans Hum Factors Electron* 8:5–15.

Eriksen, B. A., and C. W. Eriksen. 1974. Effects of noise letters upon the identification of a target letter in a non-search task. *Percept Psychophys* 16:143–6.

Fiorio, M., M. Tinazzi, and S. M. Agiloti. 2006. Selective impairment of hand mental rotation in patients with focal hand dystonia. *Brain* 129:47–54.

Fisk, J. D., and M. A. Goodale. 1985. The organization of eye and limb movements during unrestricted reaching to targets in contralateral and ipsilateral space. *Exp Brain Res* 60:159–78.

Fitts, P. M. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *J Exp Psychol* 47:381–91.

Fitts, P. M. 1964. Perceptual-motor skills learning. In *Categories of Human Learning*, ed. A. W. Melton, 243–85. New York: Academic Press.

Fitts, P. M., and C. M. Seeger. 1953. S-R compatibility: Spatial characteristics of stimulus and response codes. *J Exp Psychol* 46:199–210.

Fitts, P. M., and M. I. Posner. 1967. *Human Performance*. Belmont, CA: Brooks-Cole.

Fitts, P. M., and R. I. Deninger. 1954. S-R compatibility: Correspondence among paired elements within stimulus and response codes. *J Exp Psychol* 48:483–91.

Folk, C. L., R. W. Remington, and J. C. Johnson. 1992. Involuntary covert orienting is contingient on attentional control settings. *J Exp Psychol Hum Percept Perform* 18:1030–44.

Folk, C. L., R. W. Remington, and J. H. Wright. 1994. The structure of attentional control: Contingent attentional capture by apparent motion, abrupt onset, and color. *J Exp Psychol Hum Percept Perform* 20:317–29.

Gawryszewski, L., L. Riggio, G. Rizzolatti, and C. Umiltá. 1987. Movements of attention in the three spatial dimensions and the meaning of "neutral" cues. *Neuropsychologia* 25:19–29.

Gibson, B. S., and E. M. Kelsey. 1998. Stimulus-driven attentional capture is contingent on attentional set for display-wide visual features. *J Exp Psychol Hum Percept Perform* 24:699–706.

Heath, M. 2005. Role of limb and target vision in the online control of memory-guided reaches. *Motor Control* 9:281–311.

Hegarty, M. 2004. Mechanical reasoning as mental simulation. *Trends Cogn Sci* 8:280–5.

Helsen, W. F., D. Elliot, J. L. Starkes, and K. L. Ricker. 1998. Temporal and spatial coupling of point of gaze and hand movements in aiming. *J Mot Behav* 30:249–59.

Helsen, W. F., D. Elliott, J. L. Starkes, and K. L. Ricker. 2000. Coupling of eye, finger, elbow, and shoulder movements during manual aiming. *J Mot Behav* 32:241–8.

Hick, W. E. 1952. On the rate of gain of information. *Q J Exp Psychol* 4:11–26.

Hinton, G. E., and L. A. Parsons. 1988. Scene-based and viewer-centered representations for comparing shapes. *Cognition* 30:1–35.

Hoffman, E. R., and J. T. A. Lim. 1997. Concurrent manual-decision tasks. *Ergonomics* 40:293–318.

Holt, L. E., and S. L. Beilock. 2006. Expertise and its embodiment: Examining the impact of sensorimotor skill expertise on the representation of action-related text. *Psychon Bull Rev* 13:694–701.

Hommel, B., J. Müsseler, G. Aschersleben, and W. Prinz. 2001. The theory of event coding (TEC): A framework for perception and action planning. *Behav Brain Sci* 24:849–78.

Hornecker, E., R. J. K. Jacob, C. Hummels, B. Ullmer, A. Schmidt, E. V. D. Hoven, and A. Mazalek. 2008. TEI goes on: Tangible and embedded interaction. *IEEE Pervasive Comput* 7:91–6.

Howard, L. A., and S. P. Tipper. 1997. Hand deviations away from visual cues: Indirect evidence for inhibition. *Exp Brain Res* 113:144–52.

Hyman, R. 1953. Stimulus information as a determinant of reaction time. *J Exp Psychol* 45:188–96.

James, W. 1890. *Principles of Psychology*. New York: Holt.

Jeannerod, M. 2006. From volition to agency: The mechanism of action recognition and its failures. In *Disorders of Volition*, ed. N. Sebanz and W. Prinz, 175–92. Cambridge, MA: The MIT Press.

Jonides, J. 1981. Voluntary versus automatic control over the mind's eye's movement. In *Attention and Performance IX*, ed. J. Long and A. Baddley, 187–203. Hillsdale, NJ: Lawerence Erlbaum Assoc.

Kelso, J. A. S. 1982. The process approach to understanding human motor behavior: An introduction. In *Human Motor Behavior: An Introduction*, ed. J. A. S. Kelso, 3–19. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kelso, J. A. S. 1995. *Dynamic Patterns: The Self-organization of Brain and Behavior*. Cambridge, MA: MIT Press.

Kelso, J. A. S., D. L. Southard, and D. Goodman. 1979. On the coordination of two-handed movements. *J Exp Psychol Hum Percept Perform* 5:229–38.

Keulen, R. F., J. J. Adam, M. H. Fischer, H. Kuipers, and J. Jolles. 2002. Selective reaching: Evidence for multiple frames of reference. *J Exp Psychol Hum Percept Perform* 28:515–26.

Kilner, J. M., Paulignan, Y., Blakemore, S-J. 2003. An interference effect of observed biological movement on action. *Current Biology* 13:522–525.

Khan, M. A., I. M. Franks, D. Elliott, G. P. Lawrence, R. Chua, P. M. Bernier, S. Hansen, and D. Weeks. 2006. Inferring online and offline processing of visual feedback in target-directed movements from kinematic data. *Neurosci Biobehav Rev* 30:1106–21.

Knoblich, G., and N. Sebanz. 2006. The social nature of perception and action. *Psychol Sci* 15:99–104.

Kosslyn, S. M. 1994. *Image and Brain*. Cambridge, MA: The MIT Press.

Lindemann, O., and H. Bekkering. 2009. Object manipulation and motion perception: Evidence of an influence of action planning on visual processing. *J Exp Psychol Hum Percept Perform* 35:1062–71.

Lyons, J., D. Elliott, K. L. Ricker, D. J. Weeks, and R. Chua. 1999. Action-centred attention in virtual environments. *Can J Exp Psychol* 53:176–78.

MacKenzie, C. L., R. G. Marteniuk, C. Dugas, D. Liske, and B. Eickmeier. 1987. Three dimensional movement trajectories in Fitts' task: Implications for control. *Q J Exp Psychol* 39A:629–47.

Maltz, M., and D. Shinar. 2003. New alternative methods of analyzing human behavior in cued target acquisition. *Hum Factors* 45:281–95.

Marteniuk, R. G. 1976. *Information Processing in Motor Skills*. New York: Holt, Rinehart and Winston.

Marteniuk, R. G., C. L. MacKenzie, M. Jeannerod, S. Athenes, and C. Dugas. 1987. Constraints on human arm movement trajectories. *Can J Psychol* 41:365–78.

Marteniuk, R. G., C. L. MacKenzie, and J. L. Leavitt. 1988. Representational and physical accounts of motor control and learning: Can they account for the data? In *Cognition and Action in Skilled Behavior*, ed. A. M. Colley and J. R. Beech, 173–90. Amsterdam: Elsevier Science Publishers.

Martin, T., and D. L. Schwartz. 2005. Physically distributed learning: Adapting and reinterpreting physical environments in the development of fraction concepts. *Cogn Sci* 29:587–625.

Mazalek, A., M. Nitsche, S. Chandrasekharan, T. Welsh, P. Clifton, A. Quitmeyer, F. Peer, and F. Kirschner. 2010. Recognizing self in puppet controlled virtual avatars. In *Proceedings of Fun and Games 2010*, 66–73. New York: ACM Press.

Mazalek, A., S. Chandrasekharan, M. Nitsche, T. Welsh, G. Thomas, T. Sanka, and P. Clifton. 2009. Giving your self to the game: Transferring a player's own movements to avatars using tangible interfaces. In *Proceedings of Sandbox 2009: ACM SIGGRAPH Videogame Symposium*, 161–8. New York, NY: ACM Press.

Meegan, D. V., and S. P. Tipper. 1998. Reaching into cluttered visual environments: Spatial and temporal influences of distracting objects. *Q J Exp Psychol* 51A:225–49.

Meegan, D. V., and S. P. Tipper. 1999. Visual search and target-directed action. *J Exp Psychol Hum Percept Perform* 25:1347–62.

Meyer, D. E., R. A. Abrams, S. Kornblum, C. E. Wright, and J. E. K. Smith. 1988. Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychol Rev* 95:340–70.

Müller, H. J., and P. M. A. Rabbitt. 1989. Reflexive and voluntary orienting of visual attention: Time course of activation and resistance to interruption. *J Exp Psychol Hum Percept Perform* 15:315–30.

Nersessian, N. J. 2002. The cognitive basis of model-based reasoning in science. In *The Cognitive Basis of Science*, ed. P. Carruthers, S. Stich, and M. Siegal, 133–53. Cambridge: Cambridge University Press.

Nersessian, N. J. 2008. *Creating Scientific Concepts*. Cambridge, MA: The MIT Press.

Pashler, H. 1994. Dual-task interference in simple tasks: Data and theory. *Psychol Bull* 116:220–44.

Posner, M. I. 1982. Cumulative development of attentional theory. *Am Psychol* 37:168–79.

Posner, M. I., and Y. Cohen. 1984. Components of visual orienting. In *Attention and Performance X*, ed. H. Bouma and D. G. Bouwhuis, 531–56. Hillsdale, NJ: Lawrence Earlbaum Assoc.

Posner, M. I., and S. W. Keele. 1969. Attentional demands of movement. In *Proceedings of the 16th Congress of Applied Psychology*. Amsterdam: Swets and Zeitlinger.

Posner, M. I., M. J. Nissen, and W. C. Ogden. 1978. Attended and unattended processing modes: The role of set for spaital location. In *Modes of Perceiving and Processing Information*, ed. H. Pick and E. Saltzman, 137–57. Hillsdale, NJ: Lawrence Earlbaum Assoc.

Pratt, J., and J. McAuliffe. 2001. The effects of onsets and offsets on visual attention. *Psychol Res* 65:185–91.

Pratt, J., and R. A. Abrams. 1994. Action-centered inhibition: Effects of distractors in movement planning and execution. *Hum Mov Sci* 13:245–54.

Prinz, W. 1992. Why don't we perceive our brain states? *Eur J Cogn Psychol* 4:1–20.

Prinz, W. 1997. Perception and action planning. *Eur J Cogn Psychol* 9:129–54.

Prinz, W. 2005. A common coding approach to imitation. In *Perspectives on Imitation: From Neuroscience to Social Science*, Vol. 1, ed. S. Hurley and N. Chater, 141–56. Cambridge, MA: The MIT Press.

Proctor, R. W., and T. Van Zandt. 1994. *Human Factors in Simple and Complex Systems*. Boston: Allyn and Bacon.

Proctor, R. W., and K. L. Vu. 2003. Human information processing. In *The Human-computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emgering Applications*, ed. J. A. Jacko and A. Sears, 35–50. Hillsdale, NJ: Lawrence Earlbaum Assoc.

Rizzolatti, G., and L. Craighero. 2004. The mirror-neuron system. *Annual Reviews in Neuroscience* 27:169–92.

Rizzolatti, G., L. Riggio, J. Dascola, and C. Umilta. 1987. Reorienting attention across the horizontal and vertical meridians: Evidence in favor of a premotor theory of attention. *Neuropsychologia* 25:31–40.

Rizzolatti, G., L. Riggio, and B. M. Sheliga. 1994. Space and selective attention. In *Attention and Performance XV*, ed. C. Umiltà and M. Moscovitch, 231–65. Cambridge: MIT Press.

Rutkowski, C. 1982. An introduction to the human applications standard computer interface, part I: Theory and principles. *Byte* 7:291–310.

Shepherd, M., J. M. Findlay, and R. J. Hockey. 1986. The relationship between eye-movements and spatial attention. *Q J Exp Psychol* 38A:475–91.

Shneiderman, B. 1983. Direct manipulation: A step beyond programming languages. *IEEE Comput* 16:57–69.

Shneiderman, B. 1992. *Designing the User Interface: Strategies for Effective Human-computer Interaction*. Reading, MA: Addison-Wesley Publishing Company.

Snyder, L. H., A. P. Batista, and R. A. Andersen. 1998. Change in motor plan, without a change in the spatial locus of attention, modulates activity in posterior parietal cortex. *J Neurophysiol* 79:2814–19.

Song, J. H., and K. Nakayama. 2008. Target selection in visual search as revealed by movement trajectories. *Vis Res* 48:853–61.

Song, J. H., and K. Nakayama. 2009. Hidden cognitive states revealed in choice reaching tasks. *Trends Cogn Sci* 13:360–6.

Spence, C., D. Lloyd, F. McGlone, M. E. R. Nichols, and J. Driver. 2000. Inhibition of return is supramodal: A demonstration between all possible pairings of vision, touch, and audition. *Exp Brain Res* 134:42–8.

Stelmach, G. E. 1982. Information-processing framework for understanding human motor behavior. In *Human Motor Behavior: An Introduction*, ed. J. A. S. Kelso, 63–91. Hillsdale, NJ: Lawrence Erlbaum Associates.

Symes, E., M. Tucker, R. Ellis, L. Vainio, and G. Ottoboni. 2008. Grasp preparation improves change detection for congruent objects. *J Exp Psychol Hum Percept Perform* 34:854–71.

Telford, C. W. 1931. The refractory phase of voluntary and associative responses. *J Exp Psychol* 14:1–36.

Tenner, E. 1996. *Why Things Bite Back: Technology and the Revenge of Unintended Consequences*. New York: Alfred A. Knopf.

Tipper, S. P., C. Lortie, and G. C. Baylis. 1992. Selective reaching evidence for action-centered attention. *J Exp Psychol Hum Percept Perform* 18:891–905.

Tipper, S. P., D. Meegan, and L. A. Howard. 2002. Action-centred negative priming: Evidence for reactive inhibition. *Vis Cogn* 9:591–614.

Tipper, S. P., L. A. Howard, and G. Houghton. 1999. Behavioral consequences of selection form neural population codes. In *Attention and Performance XVIII*, ed. S. Monsell and J. Driver, 223–45. Cambridge, MA: MIT Press.

Tipper, S. P., L. A. Howard, and S. R. Jackson. 1997. Selective reaching to grasp: Evidence for distractor interference effects. *Vis Cogn* 4:1–38.

Topolinski, S., and F. Strack. 2009. Motormouth: Mere exposure depends on stimulus-specific motor simulations. *J Exp Psychol Learn Mem Cogn* 35:423–33.

Treisman, A. M. 1964a. The effect of irrelevant material on the efficiency of selective listening. *Am J Psychol* 77:533–46.

Treisman, A. M. 1964b. Verbal cues, language, and meaning in selective attention. *Am J Psychol* 77:206–19.

Treisman, A. M. 1986. Features and objects in visual processing. *Scientific Am* 255:114–25.

Treisman, A. M., and G. Gelade. 1980. A feature-integration theory of attention. *Cogn Psychol* 12:97–136.

Wallace, R. J. 1971. S-R compatibility and the idea of a response code. *J Exp Psychol* 88:354–60.

Weir, P. L., D. J. Weeks, T. N. Welsh, D. Elliott, R. Chua, E. A. Roy, and J. Lyons. 2003. Action-centred distractor effects in discrete control selection. *Exp Brain Res* 149:207–13.

Welford, A. T. 1968. *Fundamentals of Skill*. London: Methuen.

Welsh, T. N., and D. Elliott. 2004a. Movement trajectories in the presence of a distracting stimulus: Evidence for a response activation model of selective reaching. *Q J Exp Psychol* 57(A):1031–57.

Welsh, T. N., and D. Elliott. 2004b. The effects of response priming and inhibition on movement planning and execution. *J Mot Behav* 36:200–11.

Welsh, T. N., D. Elliott, J. G. Anson, V. Dhillon, D. J. Weeks, J. L. Lyons, and R. Chua. 2005. Does Joe influence Fred's action? Inhibition of return across different nervous systems. *Neurosci Lett* 385:99–104.

Welsh, T. N., D. Elliott, and D. J. Weeks. 1999. Hand deviations towards distractors: Evidence for response competition. *Exp Brain Res* 127:207–12.

Welsh, T. N., and J. Pratt. 2008. Actions modulate attentional capture. *Q J Exp Psychol* 61:968–76.

Welsh, T. N., D. J. Weeks, R. Chua, and D. Goodman. 2007. Perceptual-motor interaction: Some implications for HCI. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, 2nd ed., ed. A. Sears and J. A. Jacko, 27–41. Boca Raton, FL: CRC Press.

Welsh, T. N. and M. Zbinden. 2009. Fitts' Law and action-centered reference frames in selective reaching: The "proximity-to-hand" effect revisited. *Motor Control* 13:100–12.

Wexler, M., S. M. Kosslyn, and A. Berthoz. 1998. Motor processes in mental rotation. *Cognition* 68:77–94.

Wilson, N. L., and R. W. Gibbs. 2007. Real and imagined body movement primes metaphor comprehension. *Cogn Sci* 31:721–31.

Wohlschlager, A. 2001. Mental object rotation and the planning of hand movements. *Percept Psychophys* 63:709–18.

Woodworth, R. S. 1899. The accuracy of voluntary movements. *Psychol Rev* 3(Monograph Suppl):1–119.

Wykowska, A., A. Schubö, and B. Hommel. 2009. How you move is what you see: Action planning biases selection in visual search. *J Exp Psychol Hum Percept Perform* 35:1755–69.

Yeh, M., and C. D. Wickens. 2001a. Attentional filtering in the design of electronic map displays: A comparison of color coding, intensity coding, and decluttering techniques. *Hum Factors* 43:543–62.

Yeh, M., and C. D. Wickens. 2001b. Display signalling in augmented reality: Effects of cue reliability and image realism on attention allocation and trust calibration. *Hum Factors* 43:355–65.

Yeh, M., C. D. Wickens, and F. J. Seagull. 1999. Target cuing in visual search: The effects of conformality and display location on the allocation of visual attention. *Hum Factors* 41:524–42.

Yeh, M., J. L. Merlo, C. D. Wickens, and D. L. Bradenburg. 2003. Head up versus head down: The costs of imprecision, unreliability, and visual clutter on cue effectiveness for display signalling. *Hum Factors* 45:390–407.

Young, S. J., J. Pratt, and T. Chau. 2009. Misperceiving the speed-accuracy tradeoff: Imagined movements and perceptual decisions. *Exp Brain Res* 192:121–32.

2 Human Information Processing
An Overview for Human–Computer Interaction

Robert W. Proctor and Kim-Phuong L. Vu

CONTENTS

> It is natural for an applied psychology of human-computer interaction to be based theoretically on information-processing psychology.
>
> **—Card, Moran, and Newell (1983)**

Human–computer interaction (HCI) is fundamentally an information-processing task. When interacting with a computer or any technological device, a user has specific goals and subgoals in his or her mind. For example, smartphone users initiate the interaction by turning on or activating the device and selecting the appropriate commands needed to accomplish their desired goal. Given that the smartphone can do more than just make calls, the commands may activate applications designed to allow specific types of tasks such as playing games, e-mailing, navigating with GPS, or web surfing to be performed. The resulting output, typically displayed on the phone's screen, must provide adequate information for the user to complete the next step, or the user must enter another command to obtain the desired output. The sequence of interactions to accomplish the goals may be long and complex, and several alternative sequences, differing in efficiency, may be used to achieve these goals. During the interaction, the user is required to identify displayed information, select responses based on the displayed information, and execute those responses by entering commands. The user must search the displayed information and attend to the appropriate aspects of it. She or he must also recall the commands and the resulting consequences of those commands for different programs, remember information specific to the task that is being performed, and make decisions and solve problems during the process. For the interaction between the device and user to be efficient, the interface must be designed in accordance with the user's information-processing capabilities.

2.1 HUMAN INFORMATION-PROCESSING APPROACH

The rise of the human information-processing approach in psychology is closely coupled with the growth of the fields of cognitive psychology, human factors, and human engineering (see Proctor and Vu 2010). Although research that can be classified as falling within these fields has been conducted since the last half of the nineteenth century, their formalization dates back to World War II (see Hoffman and Defenbacher 1992). As part of the war efforts, experimental psychologists worked along with engineers on applications associated with using the sophisticated equipment being developed. As a consequence, the psychologists were exposed not only to applied problems but also to the techniques and views being developed in areas such as communications engineering (see Roscoe 2011). Many of the concepts from engineering, for instance, the notion of transmission of information through a limited capacity communications channel, were seen as applicable to analyses of human performance.

The human information-processing approach is based on the idea that human performance, from displayed information to response, is a function of several processing stages. The nature of these stages, how they are arranged, and the factors that influence how quickly and accurately a particular stage operates, can be discovered through appropriate research methods. It is often said that the central metaphor of the information-processing approach is that a human is like a computer (e.g., Lachman, Lachman, and Butterfield 1979). However, even more fundamental than the computer metaphor is the assumption that the human is a complex system that can be analyzed in terms of subsystems and their interrelation. This point is evident in the work of researchers on attention and performance, such as Paul Fitts (1951) and Donald Broadbent (1958), who were among the first to adopt the information-processing approach in the 1950s.

The systems perspective underlies not only human information processing but also human factors and HCI, providing a direct link between the basic and applied fields (Proctor and Van Zandt 2008). Human factors, in general, and HCI in particular, begin with the fundamental assumption that a human–machine system can be decomposed into machine and human subsystems, each of which can be analyzed further. The human information-processing approach provides the concepts, methods, and theories for analyzing the processes involved in the human subsystem. Posner (1986) stated, “Indeed, much of the impetus for the development of this kind of empirical study stem from the desire to integrate description of the human with overall systems” (p. V-6). Young, Clegg, and Smith (2004) emphasized that the most basic distinction among three processing stages (perception, cognition, and action), as captured in a block diagram model of human information processing, is important even for understanding the dynamic interactions of an operator with a vehicle for purposes of computer-aided augmented cognition. They note,

> “This block diagram model of the human is important because it not only models the flow of information and commands between the vehicle and the human, it also enables access to the internal state of the human at various parts of the process. This allows the modeling of what a cognitive measurement system might have access to (internal to the human), and how that measurement might then be used as part of a closed-loop human-machine interface system” (pp. 261–262).

In the first half of the twentieth century, the behaviorist approach predominated in psychology, particularly in the United States. Within this approach, many sophisticated theories of learning and behavior were developed that differed in various details (Bower and Hilgard 1981). However, the research and theories of the behaviorist approach tended to minimize the role of cognitive processes and were of limited value to the applied problems encountered in World War II. The information-processing approach was adopted because it provided a way to examine topics of basic and applied concern such as attention that were relatively neglected during the behaviorist period. It continues to be the main approach in psychology, although contributions have been made from other approaches.

Within HCI, human information-processing analyses are used in two ways. First, empirical studies evaluate the information-processing requirements of various tasks in which humans use computers. Second, computational models are developed with the intent to characterize human information processing when interacting with computers and to predict human performance with alternative interfaces. In this chapter, we survey methods used to study human information processing and summarize the major findings and the theoretical frameworks developed to explain them. We also tie the methods, findings, and theories to HCI issues to illustrate their use.

2.2 INFORMATION-PROCESSING METHODS

Any theoretical approach makes certain presuppositions and tends to favor some methods and techniques over others. Information-processing researchers have used behavioral and, to an ever-increasing extent, psychophysiological and neuroimaging measures, with an emphasis on chronometric (time-based) methods. There also has been a reliance on flow models that are often quantified through computer simulation or mathematical modeling.

2.2.1 Signal Detection Methods and Theory

One of the most useful methods for studying human information processing is that of signal detection (Macmillan and Creelman 2005). In a signal detection task, some event is classified as a signal, and the subject’s task is to detect whether the signal is present. Trials on which it is not present are called noise trials. The proportion of trials on which the signal is correctly identified as present is called the hit rate, and the proportion of trials on which the signal is incorrectly identified as present is called the false alarm rate. By using the hit and false alarm rates, whether the effect of a variable is on detectability or response bias can be evaluated.

Signal detection theory is often used as the basis for analyzing data from such tasks. This theory assumes that the response on each trial is a function of two discrete operations, encoding and decision. On a trial, the subject samples the information presented and decides whether this information is sufficient to warrant a *signal present* response. The sample of information is assumed to provide a value along a continuum of evidence states regarding the likelihood that the signal was present. The noise trials form a probability distribution of states, as do the signal trials. The decision that must be made on each trial can be characterized as whether the event is from the signal or noise distribution. The subject is presumed to adopt a criterion value of evidence above which he or she responds *signal present* and below which he or she responds *signal absent*.

In the simplest form, the distributions are assumed to be normal and equal variance. In this case, a measure of detectability, d', can be derived, as well as a measure of response bias, C (for criterion; Macmillan and Creelman 2005). The d' measure represents the difference in the means for the signal and noise distributions in standard deviation units and is found by converting the hit rate and false alarm rate to standard normal scores and obtaining the difference. A value of 0 indicates no detectability, whereas a value of 3.0 or greater indicates close to perfect detectability. The C measure is calculated by summing the standardized values of the hit and false alarm rates, and dividing by two. A value of 0 for C indicates no response bias. Positive values indicate a bias toward *signal absent* responses, and negative values indicate a bias toward *signal present* responses, with the absolute value indicating the magnitude of the bias. This measure reflects the observer's overall willingness to say *signal present*, regardless of whether it actually is present. There are numerous alternative measures of detectability and bias based on different assumptions and theories, and many task variations to which they can be applied (see Macmillan and Creelman 2005).

Signal detection analyses have been particularly useful because they can be applied to any task that can be depicted in terms of binary discriminations. For example, the proportion of words in a memory task correctly classified as old can be treated as a hit rate, and the proportion of new lures classified as old can be treated as a false alarm rate (e.g., Rotello and Macmillan 2006). In cases such as these, the resulting analysis helps researchers determine whether variables are affecting detectability of an item as old or response bias.

An area of research in which signal detection methods have been widely used is that of vigilance (Parasuraman and Davies 1977). In a typical vigilance task, a display is monitored for certain changes in it (e.g., the occurrence of an infrequent stimulus). Vigilance tasks are common in the military, but many aspects also can be found in computer-related tasks such as monitoring computer network operations (Percival and Noonan 1987). A customary finding for vigilance tasks is the vigilance decrement, in which the hit rate decreases as time on the task increases. The classic example of this vigilance decrement is that, during World War II, British radar observers detected fewer of the enemy's radar signals after 30 minutes in a radar observation shift (Mackworth 1948). Parasuraman and Davies concluded that, for many situations, the primary cause of the vigilance decrement is an increasingly strict response criterion. That is, the false alarm rate as well as the hit rate decreases as a function of time on task.

Parasuraman and Davies (1977) also provided evidence that detectability decreases across the vigil when the task requires comparison of each event to a standard held in memory and the event rate is high. Findings indicate that this decrease in detectability is a consequence of the high demand on cognitive resources imposed by such tasks. Although vigilance tasks were previously thought to be undemanding, evidence has shown that maintaining a vigil in many situations requires considerable mental effort (Warm, Parasuraman, and Matthews 2008). Our point here is that signal detection theory has played a prominent role in this research on vigilance, helping to dissociate changes in performance associated with mental demands (decreased detectability) from those due to lapses of attention (response criteria).

2.2.2 Chronometric Methods

Chronometric methods, for which time is a factor, have been the most widely used for studying human information processing. Indeed, Lachman, Lachman, and Butterfield (1979) portrayed reaction time (RT) as the main dependent measure of the information-processing approach. Although many other measures are used, RT still predominates in part because of its sensitivity and in part because of the sophisticated techniques that have been developed for analyzing RT data.

A technique called the subtractive method, introduced by Donders (1868/1969) in the 1860s, was revived in the 1950s and 1960s. This method provides a way to estimate the duration of a particular processing stage. The assumption of the subtractive method is that a series of discrete processing stages intervene between stimulus presentation and response execution. Through selection of pairs of tasks that differ by a single stage, the RT for the easier task can be subtracted from that for the more difficult task to yield the time for the additional process. Donders used three tasks hypothesized to differ with respect to stimulus identification and response selection, respectively, and estimated the time for each stage. Recently, Van de Laar et al. (2010) applied similar logic to situations in which on some trials a participant receives a "stop" signal during the reaction process, indicating that the response is to be stopped. They estimated the durations of the stop-signal identification process and a response-mapping process to be 34 and 20 ms, respectively.

The subtractive method has been used to estimate the durations of a variety of other processes, including rates of mental rotation (approximately 12–20 ms per degree of rotation; Shepard and Metzler 1971) and memory search (approximately 40 ms per item; Sternberg 1969). An application of the subtractive method to HCI would be, for example, to compare the time to find a target link on two web pages that are identical except for the number of links displayed, and to attribute the extra time to the additional visual search required for the more complex web page.

The subtractive method is only applicable when discrete, serial processing stages can be assumed. Also, the processing for the two tasks being compared must be the same except for the additional process that differentiates them. This requires an assumption of pure insertion, which is that the additional process for the more complex of two tasks can be inserted without affecting the processes held in common by the tasks. However, this assumption often is not justified.

Sternberg (1969) developed the additive factors method to allow determination of the processes involved in performing a task. The additive factors method avoids the problem of pure insertion because the crucial data are whether two variables affect RT for the same task in an additive or interactive manner. Sternberg assumed, as did Donders, that information processing occurs in a sequence of discrete stages, each of which produces a constant output that serves as input to the next stage in the sequence. With these assumptions, he showed that two variables that affect different stages should have additive effects on RT. In contrast, two variables that affect the same stage should have interactive effects on RT. Sternberg performed detailed analyses of memory search tasks in which a person holds a set of letters or digits in memory and responds to a target stimulus by indicating whether it is in the memory set. Based on the patterns of additive and interactive effects that he observed, Sternberg concluded that the processing in such tasks involves four stages: target identification, memory search, response selection, and response execution. Grobelny, Karwowski, and Drury (2005) provide an application of additive factors logic to usability of graphical icons in the design of HCI interfaces. Mode of icon array (menu or dialog box), number of icons, and difficulty of movement had additive effects on response times, implying that these variables affect different processing stages.

Both the subtractive and additive factors methods have been challenged on several grounds (Pachella 1974). First, the assumption of discrete serial stages with constant output is difficult to justify in many situations. Second, both methods rely on analyses of RT, without consideration of error rates. This can be problematic because performance is typically not error free, and, as described in Section 2.2.3, speed can be traded for accuracy. Despite these limitations, the methods have proved to be robust and useful (Sanders 1998). For example, Salthouse (2005) notes that the process analysis approach used in contemporary research into aging effects on cognitive abilities “has used a variety of analytical methods such as subtraction, additive factors … to partition the variance in the target variable into theoretically distinct processes” (p. 288).

2.2.3 Speed–Accuracy Methods

The function relating response speed to accuracy is called the speed–accuracy trade-off (Pachella 1974). The function, illustrated in Figure 2.1, shows that very fast responses can be performed with chance accuracy, and accuracy will increase as responding slows down. Of importance is the fact that when accuracy is high, as in most RT studies, a small increase in errors can result in a large decrease in RT. With respect to text entry on computing devices, MacKenzie and Soukoreff (2002) state, “Clearly, both speed and accuracy must be measured and analyzed…. Participants can enter text more quickly if they are willing to sacrifice accuracy” (pp. 159–160).

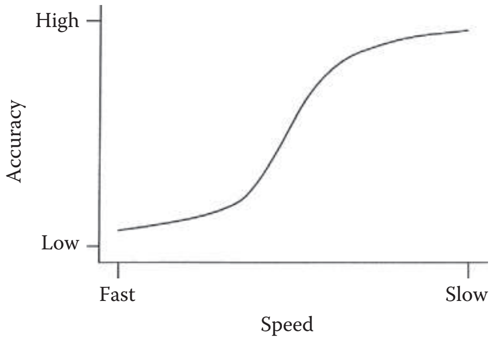


FIGURE 2.1 Speed-accuracy operating characteristic curve. Faster responding occurs at the cost of lower accuracy.

In speed–accuracy trade-off studies, the speed–accuracy criterion is varied between blocks of trials or among subjects by using different instructions regarding the relative importance of speed versus accuracy, varying payoffs such that speed or accuracy is weighted more heavily, or imposing different response deadlines (Wickelgren 1977). These studies have the potential to be more informative than RT studies because they can provide information about whether variables affect the intercept (time at which accuracy exceeds chance), asymptote (the maximal accuracy), and rate of ascension from the intercept to the asymptote, each of which may reflect different processes. For example, Boldini, Russo, and Avons (2004) obtained evidence favoring dual-process models of recognition memory over single-process models by varying the delay between a visually presented test word and a signal to respond. Recognition accuracy benefited from a modality match at study and test (better performance when the study words were also visual rather than auditory) at short response-signal delays, but it benefited from deep processing during study (judging pleasantness) over shallow processing (repeating aloud each word) at long response-signal delays. Boldini et al. interpreted these results as consistent with the view that recognition judgments are based on a fast familiarity process or a slower recollection process.

In tasks requiring search of complex visual displays, a speed emphasis may influence more than just the criterion for emitting a response. McCarley (2009) had young adults perform a simulated baggage-screening task under instructions that emphasized speed or accuracy of responding. With speed emphasis, the participants made fewer eye fixations of shorter duration than under accuracy emphasis. Reduction in accuracy was a consequence mainly of failure to fixate the target of the search rather than a failure to respond to targets that were fixated. This study illustrates how a speed–accuracy trade-off manipulation can be of value in applied contexts.

Because the speed–accuracy criterion is manipulated in addition to any other variables of interest, much more data must be collected in a speed–accuracy study than in a typical RT study. Consequently, use of speed–accuracy methods has been restricted to situations in which the speed–accuracy relation is of major concern or of apparent significant value, rather than being widely adopted as the method of choice.

2.2.4 Psychophysiological and Neuroimaging Methods

In the past decade, psychophysiological and neuroimaging methods have been used increasingly to evaluate implications of information-processing models and to relate the models to brain processes. This area of research is called *cognitive neuroscience* (Ward 2010). Such methods can provide details regarding the nature of processing by examining physiological activity as a task is being performed. The most widely used psychophysiological method involves measurement of electroencephalograms (EEGs), which are recordings of changes in brain activity as a function of time as measured from electrodes placed on the scalp (Rugg and Coles 1995). Different frequency bands of EEG rhythms can be distinguished that can be related to subjective states and the processes underlying task performance.

One application of EEGs to HCI in recent years has been the development of brain–computer interfaces that allow a person to control technological devices through the use of brain signals. Such interfaces are of value for motor-disabled persons who are not able to communicate through traditional data-entry devices. Changes in EEGs that arise from different types of mental processing can be coded into distinct computer commands, and people can be trained to use their thoughts to control the computer's interface (e.g., Kauhanen et al. 2007). This mode of HCI opens up possibilities for disabled persons to interact with their environment and communicate with other people.

Of most concern for information-processing research are event-related potentials (ERPs), which are the changes in brain activity that are elicited by an event such as stimulus presentation or response initiation. ERPs are obtained by averaging across many trials of a task to remove background EEG noise and are thought to reflect postsynaptic potentials in the brain. There are several features of the ERP that represent different aspects of processing. These features are labeled according to their polarity, positive (P) or negative (N), and their sequence or latency. The first positive (P1) and negative (N1) components are associated with early perceptual processes. They are called exogenous components because they occur in close temporal proximity to the stimulus event and have a stable latency with respect to it. Later components reflect cognitive processes and are called endogenous because they are a function of the task demands and have a more variable latency than the exogenous components. One such component that has been studied extensively is the P3 (or, P300), which represents postperceptual processes. When an occasional target stimulus is interspersed in a stream of standards, the P3 is observed in response to targets, but not to standards. By comparing the effects of task manipulations on various ERP components such as P3, their onset latencies, and their scalp distributions, relatively detailed inferences about the cognitive processes can be made.

An early application of P3 analysis to HCI is a study by Trimmel and Huber (1998). In their study, subjects performed three HCI tasks (text editing, programming, and playing the game *Tetris*) for 7 minutes each. They also performed comparable paper/pencil tasks in three other conditions. The P3 was measured after each experimental task by having subjects monitor a stream of high- and low-pitched tones, keeping count of each separately. The P3 varied as a function of type of task, as well as medium (computer vs. paper/pencil). The amplitude of the P3 was smaller following the HCI tasks than following the paper/pencil tasks, suggesting that the HCI tasks caused more fatigue or depletion of cognitive resources than the paper/pencil task. The P3 latency was shorter after the programming task than after the others, which the authors interpreted as an aftereffect of highly focused attention.

Another measure that has been used in studies of human information processing is the lateralized readiness potential (LRP; Eimer 1998). The LRP can be recorded in choice-reaction tasks that require a response with the left or right hand. It is a measure of differential activation of the lateral motor areas of the visual cortex that occurs shortly before and during execution of a response. The asymmetric activation favors the motor area contralateral to the hand making the response, because this is the area that controls the hand. The LRP has been obtained in situations in which no overt response is ever executed, allowing it to be used as an index of covert, partial response activation. The LRP is thus a measure of the difference in activity from the two sides of the brain that can be used as an indicator of covert reaction tendencies, to determine whether a response has been prepared even when it is not actually executed. It can also be used to determine whether the effects of a variable are before or subsequent to response preparation.

Electrophysiological measurements do not have the spatial resolution needed to provide precise information about the brain structures that produce the recorded activity, although advances in the technology are producing continual improvements in this regard. Much work has been done recently, though, on neuroimaging methods that provide better spatial resolution. These include positron-emission tomography, functional magnetic resonance imaging (fMRI), and transcranial Doppler sonography, which measure changes in blood flow associated with neuronal activity in different regions of the brain (Huettel, Song, and McCarthy 2004). Traditionally, these methods have poorer temporal resolution than the electrophysiological methods, but with the introduction of more sophisticated techniques, the gap in temporal resolution has been greatly reduced.

In an imaging study, often both control and experimental tasks are performed, and the functional neuroanatomy of the cognitive processes is derived by subtracting the image during the control task from that during the experimental task. This subtractive method of neuroimaging analysis has the same limitations as that for reaction-time analysis (Sartori and Umiltà 2000). Stevenson, Kim, and James (2009) provided evidence that an additive factors analysis of fMRI data, in which interactive vs. additive effects of different independent variables are compared, "provides a method for investigating multisensory interactions that goes beyond what can be achieved with more established metric-based, subtraction-type methods" (p. 183).

Application of cognitive neuroscience to human factors and HCI has been advocated under the heading of neuroergonomics (e.g., Lees et al. 2010). According to Parasuraman (2003), "Neuroergonomics focuses on investigations of the neural bases of mental functions and physical performance in relation to technology, work, leisure, transportation, health care and other settings in the real world" (p. 5). Neuroergonomics has the goal of using knowledge of the relation between brain function and human performance to design interfaces and computerized systems that are sensitive to brain function with the intent of increasing the efficiency and safety of human–machine systems.

2.3 INFORMATION-PROCESSING MODELS

It is common to assume that the processing between stimuli and responses consists of a series of discrete stages for which the output for one stage serves as the input for the next, as Donders and Sternberg assumed. This assumption is made for the Model Human Processor (Card, Moran, and Newell 1983) and the Executive-Process Interactive Control (EPIC; Meyer and Kieras 1997) architectures, among others, both of which have been applied to HCI. However, models can be developed that allow for successive processing stages to operate concurrently. McClelland's (1979) cascade model, in which partial information at one subprocess, or stage, is transferred to the next, is of this type. Each stage is continuously active, and its output is a continuous value that is always available to the next stage. The final stage results in selection of which of the possible alternative responses to execute. Many parallel distributed processing, or neural network, models are of a continuous nature.

According to J. Miller (1988), models of human information processing can be classified as discrete or continuous along three dimensions: representation, transformation, and transmission. Representation refers to whether the input and output codes for the processing stage are continuous or discrete. Transformation refers to whether the operation performed by the processing stage (e.g., spatial transformation) is continuous or discrete. Transmission is classified as discrete if the processing of successive stages does not overlap temporally. The discrete stage model proposed by Sternberg (1969) has discrete representation and transmission, whereas the cascade model proposed by McClelland (1979) has continuous representation, transmission, and transformation. Models can be intermediate to these two extremes. For example, Miller's (1988) asynchronous discrete coding model assumes that most stimuli are composed of features, and these features are identified separately. Discrete processing occurs for feature identification, but once a feature is identified, this information can be passed to response selection while the other features are still being identified.

Sequential sampling models are able to account for both RT and accuracy, and consequently, the trade-off between them (Ratcliff and Smith 2004; Van Zandt, Colonius, and Proctor 2000). Such models are dynamic models of signal detection, in which decisions are based on a series of samples from the probability distributions rather than a single sample. Each sample is classified as favoring one alternative or another, and this information is fed into a decision mechanism in which gradual accumulation of the information occurs until a response threshold is reached, at which time that response is made. As Busemeyer and Diederich (2010) note, "Dynamic models of signal detection have proven to be very effective for simultaneously analyzing choice probability and the response time distributions for signal detection tasks" (p. 89).

Various types of the dynamic models have been developed and applied to an array of experimental tasks. In such models, factors that influence the quality of information processing (the detectability or discriminability) have their effects on the rate at which the information accumulates. In contrast, factors that bias speed versus accuracy or factors that produce biases toward particular responses have their effects on the response thresholds.

Sequential sampling can be incorporated into more complete cognitive architectures to model speed and accuracy. These architectures specify properties of various processing stages and stores, such as memory and decision processes, and provide a means for developing specific models to simulate performance of a range of tasks. One widely used architecture of this type is Adaptive Control of Thought—Rational (Anderson et al. 2004), which has been used to model, for example, improvements in performance and retention with practice (practice and retention [Anderson, Fincham, and Douglass 1999] and the choices in decision-making tasks [Gonzalez, Lerch, and Lebiere 2003]).

2.4 INFORMATION PROCESSING IN CHOICE-REACTION TASKS

In a typical choice-reaction task in which each stimulus is assigned to a unique response, it is customary to distinguish between three stages of processing: stimulus identification, response selection, and response execution (Proctor and Van Zandt 2008). The stimulus-identification stage involves processes that are entirely dependent on stimulus properties. The response-selection stage concerns those processes involved in determining which response to make to each stimulus. Response execution refers to programming and execution

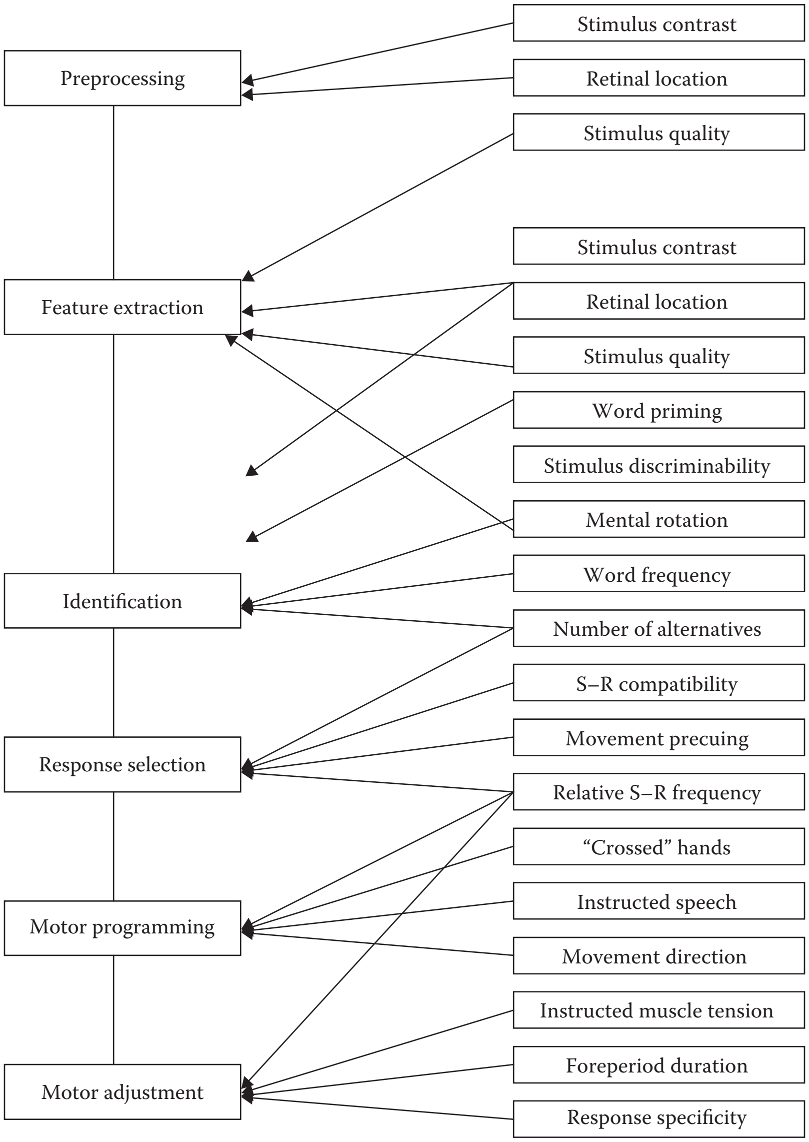


FIGURE 2.2 Information-processing stages and variables that affect them, based on Sanders' (1998) taxonomy. (From Sanders, A. F., *Elements of Human Performance*, Erlbaum, Mahwah, New Jersey, 1998. With permission.)

of motor responses. Based on additive factors logic, Sanders (1998) decomposed the stimulus-identification stage into three subcategories and the response-execution stage into two subcategories, resulting in six stages (see Figure 2.2).

2.4.1 Stimulus Identification

The preprocessing stage of stimulus identification refers to peripheral sensory processes involved in the conduction of the sensory signal along the afferent pathways to the sensory projection areas of the cerebral cortex. These processes are affected by variables such as stimulus contrast and retinal location. As stimulus contrast, or intensity, decreases, RT increases. For example, Miles and Proctor (2009) had participants make left and right keypress responses to the nonspatial or spatial feature of centrally presented location words. The discriminability of the spatial feature of the word, or of both the spatial and nonspatial features, was manipulated. When the spatial feature of the word was task-irrelevant, decreasing the discriminability of this feature reduced the typical benefit for correspondence of the word meaning with the key press response to the relevant stimulus feature. This correspondence benefit was restored when the discriminability of both the task-relevant and task-irrelevant features were reduced together, slowing the processing of both the relevant and irrelevant information. These results suggest that reduction of discriminability slows processing of the perceptual information but does not alter the response-selection processes that operate on that information.

Feature extraction involves lower-level perceptual processing based in area V1 (the visual cortex) and other early visual cortical areas. Stimulus discriminability, word priming, and stimulus quality affect the feature extraction process. For example, manipulations of stimulus quality such as superimposing a grid slow RT, presumably by creating difficulty for the extraction of features. Identification itself is influenced by word frequency and mental rotation. The latter refers to that when a stimulus is rotated from the upright position, the time it takes to identify the stimulus increases as an approximately linear function of angular deviation from upright (Shepard and Metzler 1971; see also Section 2.2.2). This increase in identification time is presumed to reflect a normalization process by which the image is mentally rotated in a continuous manner to the upright position.

2.4.2 Response Selection

Response selection refers to those processes involved in determining what response to make to a particular stimulus. It is affected by the number of alternatives, stimulus–response compatibility, and precuing (providing advance information about a forthcoming event). RT increases as a logarithmic function of the number of stimulus–response alternatives (Hick 1952; Hyman 1953). This relation is known as the Hick–Hyman law, which for *N* equally likely alternatives is as follows:

$$\mathrm{RT} = a + b \log_2 N \tag{2.1}$$

where *a* is the base processing time, and *b* is the amount that RT increases with increases in *N*. The slope of the Hick–Hyman function is influenced by many factors. For example, the slope decreases as subjects become practiced at a task (Teichner and Krebs 1974). Usher, Olami, and McLelland (2002) provided evidence from fits of a sequential sampling model that the Hick–Hyman law is due to subjects' adjusting their response criteria upward as the number of alternatives increases, in an attempt to maintain a constant high level of accuracy.

One variable that influences the slope of the Hick–Hyman function is stimulus–response compatibility, which has considerable impact on response-selection efficiency (see Proctor and Vu 2006, for a review of compatibility principles). Compatibility effects are differences in speed and accuracy of responding as a function of how natural, or compatible, the relation between stimuli and responses is. Two types of compatibility effects can be distinguished (Kornblum, Hasbroucq, and Osman 1990). For one type, certain sets of stimuli are more compatible with certain sets of responses than with others. For example, the combinations of verbal–vocal and spatial–manual sets yield better performance than the combinations of verbal–manual and spatial–vocal sets (Wang and Proctor 1996). For the other type, within a specific stimulus–response set, some mappings of individual stimuli to responses produce better performance than others. If one stimulus has the meaning "left" and the other "right," performance is better if the left stimulus is mapped to the left response and the right stimulus to the right response, for all stimulus and response modes.

Fitts and Seeger (1953) and Fitts and Deininger (1954) demonstrated both types of compatibility effects for spatially arranged display and response panels. However, compatibility effects occur for a much wider variety of other stimulus–response sets. According to Kornblum, Hasbroucq, and Osman (1990), dimensional overlap (similarity) between the stimulus and response sets is the critical factor. When the sets have dimensional overlap, a stimulus will activate its corresponding response automatically. If this response is correct (compatible mapping), responding will be facilitated, but if it is not correct (incompatible mapping), responding will be inhibited. A second factor contributing to the advantage for the compatible mapping is that intentional translation of the stimulus into a response will occur quicker when the mapping is compatible than when it is not. Most contemporary models of stimulus–response compatibility include both automatic and intentional response-selection routes (Hommel and Prinz 1997), although they differ regarding the exact conditions under which each plays a role and the way in which they interact.

One reason why automatic activation is considered to contribute to compatibility effects is that such effects occur when irrelevant stimulus information overlaps with the response set (Lu and Proctor 1995). The Stroop color-naming effect, for which an incongruent color word produces interference in naming a relevant stimulus color, is most well-known example. An irrelevant stimulus location also produces interference when it is incongruent with the location of a key press to a relevant stimulus dimension, a phenomenon known as the Simon effect (Simon 1990). Psychophysiological studies in which the LRP has been measured have provided evidence that the Simon effect is due, at least in part, to activation of the response corresponding to stimulus location (Melara et al. 2008).

For completely unrelated stimulus and response sets that are structured, performance is better when structural correspondence is maintained (Reeve and Proctor 1990). For instance, when stimuli and responses are ordered (e.g., a row of four stimulus locations and a row of four response locations), RT is faster when the stimulus–response mapping can be characterized by a rule (e.g., press the key at the mirror opposite location) than when the mapping is random (Duncan 1977). Spatial compatibility effects also occur when display and response elements refer to orthogonal spatial dimensions (Proctor and Cho 2006). However, stimulus–response compatibility effects sometimes do not occur under conditions in which one would expect them to. For example, when compatible and incompatible mappings are mixed within a single block, the typical compatibility effect is eliminated (Shaffer 1965; Vu and Proctor 2004). Moreover, the same display and response elements can be coded along multiple dimensions in certain situations (e.g., vertical position vs. horizontal position). The relative importance of maintaining compatibility on each dimension is a function of how salient the dimensions are made by the task environment (Rubichi et al. 2006).

Stimulus–response compatibility effects occur for older adults as well as younger adults, with older adults typically showing larger compatibility effects that cannot be attributed entirely to general slowing (Proctor, Vu, and Pick 2005). Although older adults show a greater cost of incompatibility than do younger adults, evidence indicates that the processing of information proceeds in a similar, though slower, manner (Vu and Proctor 2008). Because the older adults' response times increase disproportionally as a function of uncertainty, they benefit more from a precue that either indicates which of two tasks will be performed or reduces the number of possible stimulus and response alternatives (Vu and Proctor 2008). Implications of these findings for HCI are that older adults' performance will suffer more from incompatibility in designs, but this cost can be minimized by design strategies that limit the amount of information that must be processed.

Responses often produce effects in the environment, as, for example, when flipping a switch turns on a light. Studies have shown that speed of response selection is also influenced by such response-effect compatibility. Kunde (2001) had participants respond to the color of a single stimulus centered on a display screen by pressing one of four response keys, arranged in a row, with the index and middle fingers of the hands. Pressing a key filled in one box in a row of four outline boxes located above the response keys. Performance was faster when the mapping of keys to the filled-in boxes was spatially compatible than when it was incompatible. That response time is influenced by compatibility of a response with the effect that it produces implies that actions are selected and performed in anticipation of their consequences.

Because situations in which compatibility effects will influence performance are not always obvious, interface designers may make poor decisions if they rely only on their intuitions. Payne (1995), Vu and Proctor (2003), and Tlauka (2004) showed that naïve subjects can predict basic compatibility effects such as that performance will be better with a mapping that is spatially compatible than with one that is not. However, they do not accurately predict many other compatibility effects that occur such as the benefit of maintaining a consistent stimulus–response mapping rule. One encouraging finding is that estimates of relative compatibility can be improved by a small amount of experience performing with the different stimulus–response mappings (Vu and Proctor 2003). Designers need to be aware of the potential problems created by various types of incompatibility between display and response elements because their influences are not always obvious. A designer can get a better feel for the relative compatibility of alternative arrangements by performing tasks that use them. However, after the designer selects a few arrangements that would seem to yield good performance, more thorough usability testing of the remaining arrangements on groups of users needs to be performed.

2.4.3 Response Execution

Motor programming refers to specification of the physical response that is to be made. This process is affected by variables such as relative stimulus–response frequency and movement direction. One factor that influences this stage is movement complexity. The longer the sequence of movements that is to be made upon occurrence of a stimulus in a choice-reaction task, the longer the RT to initiate the sequence (Sternberg et al. 1978). This effect is thought to be due to the time required to load the movement sequence into a buffer before initiating the movements. Time to initiate the movement sequence decreases with practice, and fMRI evidence suggests that this decrease in RT involves distinct neural systems that support visuomotor learning of finger sequences and spatial learning of the locations of the finger movements on a keypad (Parsons, Harrington, and Rao 2005).

One of the most widely known relations attributed to response execution is Fitts's law, which describes the time to make aimed movements to a target location (Fitts 1954). This law, as originally specified by Fitts, is as follows:

$$\text{Movement Time} = a + b \log_2(2D / W) \qquad (2.2)$$

where a and b are constants, D is distance to the target, and W is target width. However, there are slightly different versions of the law. According to Fitts's law, movement time is a direct function of distance and an inverse function of target width. Fitts's law has been found to provide an accurate description of movement time in many situations, although alternatives have been proposed for certain situations. One factor that contributes to the increase in movement time as the index of difficulty increases is the need to make a corrective submovement based on feedback in order to hit the target location (Meyer et al. 1988).

The importance of Fitts's law for HCI is illustrated by the fact that the December 2004 issue of the *International Journal of Human-Computer Studies* was devoted to the fiftieth anniversary of Fitts's original study. In the preface to the issue, the editors, Guiard and Beudouin-Lafon (2004), state, "What has come to be known as Fitts's law has proven highly applicable in Human–Computer Interaction (HCI), making it possible to predict reliably the minimum time for a person in a pointing task to reach a specified target" (p. 747). Several illustrations of this point follow.

One implication of the law for interface design is that the slope of the function, b, may vary across different control devices, in which case, movement times will be faster for the devices that yield lower slopes. Card, English, and Burr (1978) conducted a study that evaluated how efficient text keys, step keys, a mouse, and a joystick are at a text-selection task, in which users selected text by positioning the cursor on the desired area and pressing a button or key. They showed that the mouse was the most efficient device for this task: Positioning time for the mouse and joystick could be accounted for by Fitts's law, with the slope of the function being less steep for the mouse; positioning time with the keys was proportional to the number of key strokes that had to be executed.

Another implication of Fitts's law is that any reduction in the index of difficulty should decrease the time for movements. Walker, Smelcer, and Nilsen (1991) evaluated movement time and accuracy of menu selection for the mouse. Their results showed that reducing the distance to be traveled (which reduces the index of difficulty) by placing the initial cursor in the middle of the menu, rather than the top, improved movement time. Placing a border around the menu item in which a click would still activate that item, and increasing the width of the border as the travel distance increases, also improved performance. The reduction in movement time by use of borders is predicted by Fitts's law because borders increase the size of the target area. McGuffin and Balakrishnan (2005) showed that a similar reduction in movement time can be accomplished by expanding the target size while the movement is taking place.

Gillan et al. (1992) noted that designers must be cautious when applying Fitts's law to HCI because factors other than distance and target size play a role when using a mouse. Specifically, they proposed that the critical factors in pointing and dragging are different than those in pointing and clicking (which was the main task in Card, English, and Burr [1978] study). Gillan et al. showed that, for a text-selection task, both point–click and point–drag movement times can be accounted for by Fitts's law. For point–click sequences, the diagonal distance across the text object, rather than the horizontal distance, provided the best fit for pointing time. For point–drag, the vertical distance of the text provided the best fit. The reason why the horizontal distance is irrelevant is that the cursor must be positioned at the beginning of the string for the point–drag sequence. Thus, task requirements should be considered before applying Fitts's law to the interface design.

Motor adjustment deals with the transition from a central motor program to peripheral motor activity. Studies of motor adjustment have focused on the influence of foreperiod duration on motor preparation. In a typical study, a neutral warning signal is presented at various intervals before the onset of the imperative stimulus. Bertelson (1967) varied the duration of the warning foreperiod and found that RT reached a minimum at a foreperiod of 150 ms and then increased slightly at 200- and 300-ms foreperiods. However, error rate increased to a maximum at the 150-ms foreperiod and decreased slightly at the longer foreperiods. This relatively typical pattern suggests that it takes time to attain a state of high motor preparation, and that this state reflects an increased readiness to respond quickly at the expense of accuracy.

2.5 MEMORY IN INFORMATION PROCESSING

Memory refers to explicit recollection of information in the absence of the original stimulus and to persisting effects of that information on information processing that may be implicit. Memory may involve recall of an immediately preceding event or one many years in the past, knowledge derived from everyday life experiences and education, or procedures learned to accomplish complex perceptual-motor tasks. Memory can be classified into several categories. Episodic memory refers to memory for a specific event such as going to the movie last night, whereas semantic memory refers to general knowledge such as what a movie is. Declarative memory is verbalizable knowledge, and procedural memory is knowledge that can be expressed nonverbally. In other words, declarative memory is knowing that something is the case, whereas procedural memory is knowing how to do something. For example, telling your friend your new phone number involves declarative memory, whereas riding a bicycle involves procedural knowledge. A memory test is regarded as explicit if a person is asked to judge whether a specific item or event has occurred before in a particular context; the test is implicit if the person is to make a judgment, such as whether a string of letters is a word or nonword, that can be made without reference to earlier "priming" events. In this section, we focus primarily on explicit episodic memory.

Three types of memory systems are customarily distinguished: sensory stores, short-term memory (STM; or working memory), and long-term memory (LTM). Sensory stores, which we will not discuss in detail, refer to brief modality-specific persistence of a sensory stimulus from which information can be retrieved for 1 or 2 seconds (see Nairne 2003). STM and LTM are the main categories by which investigations of episodic memory are classified, and as the terms imply, the distinction is based primarily on duration. The dominant view is that these are distinct systems that operate according to different principles, but there has been debate over whether the processes involved in these two types of memories are the same or different. An fMRI study by Talmi et al. (2005) found that recognition of early items in the list was accompanied by activation of areas in the brain associated with LTM, whereas recognition of recent items did not, supporting a distinction between STM and LTM stores.

2.5.1 Short-Term (Working) Memory

STM refers to representations that are currently being used or have recently been used and last for a short duration. A distinguishing characteristic is that STM is of limited capacity. This point was emphasized in Miller's (1956) classic article, "The Magical Number Seven Plus or Minus Two," in which he indicated that capacity is not simply a function of the number of items, but rather the number of "chunks." For example, "i, b, m" are three letters, but most people can combine them to form one meaningful chunk of "IBM." Subsequent evidence indicates that the capacity of STM for verbal material is less than originally estimated by Miller, being three chunks when covert rehearsal is prevented (Chen and Cowan 2009). As a consequence of chunking, memory span is similar for strings of unrelated letters and strings of meaningful acronyms or words. Researchers refer to the number of items that can be recalled correctly, in order, as memory span. When rehearsal is not prevented, the memory span for words varies as a function of word length: The number of words that can be retained decreases as word length increases (Baddeley, Thomson, and Buchanan 1975). Evidence has indicated that the capacity is the number of syllables that can be said in about 2 seconds (Schweickert and Boruf 1986).

As most people are aware from personal experience, if distracted by another activity, information in STM can be forgotten quickly. With respect to HCI, Oulasvirta and Saariluoma (2004) note that diversion of attention from the current task to a competing task is a common occurrence, for example, when an unrequested pop-up dialog box requiring an action appears on the screen. Laboratory experiments have shown that recall of a string of letters that is within the memory span decreases to close to chance levels over a retention interval of 18 seconds when rehearsal is prevented by an unrelated distractor task (Brown 1958; Peterson and Peterson 1959). This short-term forgetting was thought initially to be a consequence of decay of the memory trace due to prevention of rehearsal. However, Keppel and Underwood (1962) showed that proactive interference from items on previous lists is a significant contributor to forgetting. They found no forgetting at long retention intervals when only the first list in a series was examined, with the amount of forgetting being much larger for the second and third lists as proactive interference built up. Consistent with this interpretation, "release" from proactive inhibition, that is, improved recall, occurs when the category of the to-be-remembered items on the current list differs from that of previous lists (Wickens 1970).

As the complexity of an HCI task increases, one consequence is to overload STM. Jacko and Ward (1996) varied four different determinants of task complexity (multiple paths, multiple outcomes, conflicting interdependence among paths, or uncertain or probabilistic linkages) in a task requiring use of a hierarchical menu to acquire specified information. When one determinant was present, performance was slowed by approximately 50%, and when two determinants were present in combination, performance was slowed further. That is, as the number of complexity determinants in the interface increased, performance decreased. Jacko and Ward attributed the decrease in performance for all four determinants to the increased STM load they imposed.

The best-known model of STM is Baddeley and Hitch's (1974) working memory model, which partitions STM into three main parts: central executive, phonological loop, and visuospatial sketchpad. The central executive is closely tied to the focus of attention. It is involved in computational processing, as in performing mental arithmetic, as well as in controlling and coordinating the actions of the phonological loop and visuospatial sketchpad. The phonological loop is composed of a phonological store that is responsible for storage of the to-be-remembered items, and an articulatory control process that is responsible for recoding verbal items into a phonological form and rehearsal of those items. The items stored in the phonological store decay over a short interval and can be refreshed through rehearsal from the articulatory control process. The visuospatial sketchpad retains information regarding visual and spatial information, and it is involved in mental imagery.

The working memory model has been successful in explaining several phenomena of STM (Baddeley 2000; 2003). However, the model cannot explain why memory span for visually presented material is only slightly reduced when subjects engage in concurrent articulatory suppression (such as saying the words "the" aloud repeatedly). Articulatory suppression should monopolize the phonological loop, preventing any visual items from entering it. To account for such findings, Baddeley revised the working memory model to include an episodic buffer (see Figure 2.3). The buffer is a limited capacity temporary store that can integrate information from the phonological loop, visuospatial sketchpad, and LTM. By attending to a given source of information in the episodic buffer, the central executive can create new cognitive representations that might be useful in problem solving.

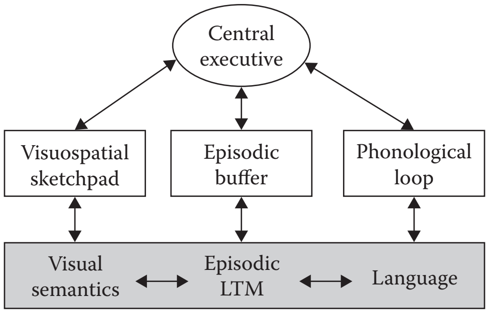


FIGURE 2.3 Baddeley's (2000) revised working memory model. (Reprinted from *Trends in Cogn Sci*, 4, Baddeley, A. D., The episodic buffer: A new component of working memory? 421, Copyright 2000, with permission from Elsevier.)

2.5.2 Long-Term Memory

LTM refers to representations that can be remembered for durations longer than can be attributed to STM. LTM can involve information presented minutes ago or years ago. Initially, it was thought that the probability of an item being encoded into LTM was a direct function of the amount of time that it was in STM, or how much it was rehearsed. However, Craik and Watkins (1973) showed that rehearsal in itself is not sufficient, but rather that deep-level processing of the meaning of the material is the important factor in transferring items to LTM. They presented subjects with a list of words and instructed them that when the experimenter stopped the presentation, they were to recall the last word starting with the letter "a." The number of other words between instances of "a" words was varied with the idea that the amount of time a word was rehearsed would depend on the number of words before the next "a" word. At the end of the session, subjects were given a surprise test in which they were to recall all "a" words. There was no effect of number of intervening words on recall, suggesting that although subjects rehearsed the words longer, their recall did not improve because the words were not processed deeply.

Craik and Watkins' (1973) results are consistent with the levels of processing framework proposed by Craik and Lockhart (1972). According to this view, encoding proceeds in a series of analyses, from shallow perceptual features to deeper, semantic levels. The deeper the level of processing, the more strongly the item is encoded in memory. A key study supporting the levels of processing view is that of Hyde and Jenkins (1973). In their study, groups of subjects were presented a list of words for which they engaged in shallow processing (e.g., deciding whether each word contained a capital letter) or deep processing of it (e.g., identifying whether each word was a verb or a noun). Subjects were not told in advance that they would be asked to recall the words, but were given a surprise recall test at the end of the session. Results showed that the deep processing group recalled more words than the shallow processing group. Of direct relevance to HCI, Oulasvirta, Kärkkäinen, and Laarni (2005) found that participants who viewed the content area of a web page had no better memory for the material than that guessed by a control group who had never seen the page, because the participants' task was to locate links on the page and not to process the content information.

Another well-known principle for LTM is encoding specificity, which states that the probability that a retrieval cue results in recollection of an earlier event is an increasing function of the match between the features encoded initially and those provided by the retrieval cue (Surprenant and Neath 2009). An implication of this principle is that memory will be context dependent. Godden and Baddeley (1975) demonstrated a context-dependent memory effect by having divers learn a list of words on land or under water, and recall the words on land or under water. Recall was higher for the group who learned on land when the test took place on land than under water, and vice versa for the group who learned under water. A related principle is that of transfer appropriate processing (Morris, Bransford, and Franks 1977). Morris et al. showed that deep-level semantic judgments during study produced better performance than shallow rhyme judgments on a standard recognition memory test. However, when the memory test required decisions about whether the test words rhymed with studied words, the rhyme judgments led to better performance than the semantic judgments. Brain imaging evidence consistent with transfer appropriate processing was obtained by Park and Rugg (2008), who found that word and picture stimuli on a recognition memory test produced greater activity in brain regions associated with those stimulus modes when the original study stimuli were also presented in the same mode. Healy, Wohldman, and Bourne (2005) have proposed that encoding specificity and transfer appropriate processing can be incorporated within the single principle of procedural reinstatement: Retention will be evident to the extent that the procedures engaged in during study or training are reinstated at the retention test.

Research has confirmed that the levels of processing framework must accommodate the effects of the retention context, as captured by the above principles, to explain the effects of processing performed during encoding. Although levels-of-processing has a strong effect on accuracy of explicit recall and recognition, Jacoby and Dallas (1981) found no effect on an implicit memory test. Later studies have shown a robust effect of levels-of-processing on implicit tests similar to that obtained for recall and recognition if the test is based on conceptual cues, rather than perceptual cues (Lee 2008).

2.5.3 Other Factors Affecting Retrieval of Earlier Events

Memory researchers have studied many factors that influence long-term retention. Not surprisingly, episodic memory improves with repetition of items or events. Also, massed repetition (repeating the same item in a row) is less effective than spaced repetition (repeating the same item with one or more intervening items). This benefit for spaced repetition, called the spacing effect or lag effect, is often attributed to two main factors. First, study time for the same items appearing in succession is less than study time for the same items appearing further apart. Second, when the items are studied over a longer period of time, there is an opportunity for the items to be associated with different cues that can aid recall later. The spacing or lag effect is widespread and occurs for both recall and recognition (Hintzman 1974). Bahrick and Hall (2005) noted that a similar spacing benefit is found for learning lists of items when practice sessions, each with test and learning phases, are separated by several days. They presented evidence that a large part of the spacing benefit in this case arises from individuals determining which study strategies are more effective at promoting long-term retention and then using those strategies more.

Another widely studied phenomenon is the generation effect, in which recall is better when subjects have to generate the to-be-remembered words rather than just studying the words as they are presented (Slamecka and Graf 1978). In a generation effect experiment, subjects are divided into two groups: read and generate. Each group receives a series of words, with each word spelled out completely for the read group and missing letters for the generate group. An example is as follows:

Read group: CAT; ELEPHANT; GRAPE; CAKE
Generate group: C _ T; E_E_H _ NT; G _ APE; CAK_

The typical results show that subjects in the generate group can recall more words than those in the read group. One application of the generation effect to HCI is that when a computer user needs a password for an account, the system should allow the user to generate the password rather than providing him or her with one because the user would be more likely to recall the generated password. The common method of proactive password generation, in which users are asked to generate a password that meets certain restrictions (e.g., contain an uppercase letter, a lowercase letter, a digit, etc.), is intended to result in more memorable and secure passwords (see, e.g., Vu et al. 2007).

Events that precede or follow an event of interest can interfere with recall of that event. The former is referred to as proactive interference, and was discussed in the section on STM, and the latter is referred to as retroactive interference. One area of research in which retroactive interference is of central concern is that of eyewitness testimony. Loftus and Palmer (1974) showed that subsequent events could distort a person's memory of an event that the person witnessed. Subjects were shown a sequence of events depicting a car accident. Subsequently, they were asked the question, "How fast were the cars going when they _____ each other." When the verb "contacted" was used, subjects estimated the speed to be 32 mph, and only one-tenth of them reported seeing broken glass. However, when the verb "smashed" was used, the estimated speed increased to 41 mph, and almost one-third of the subjects reported seeing broken glass. Demonstrations like these indicate not only that retroactive interference can cause forgetting of events, but that it also can cause the memory of events to be changed. More recent research has shown that completely false memories can be implanted (see Roediger and McDermott 1995).

Mnemonic techniques can also be used to improve recall. The basic idea behind mnemonics is to connect the to-be-remembered material with an established organizational structure that can be easily accessible later on. Two widely used mnemonic techniques are the pegword method (Wood and Pratt 1987) and the method of loci (Verhaeghen and Marcoen 1996). In the pegword method, a familiar rhyme provides the organizational structure. A visual image is formed between each pegword in the rhyme and the associated target item. At recall, the rhyme is generated, and the associated items come to mind. For the method of loci, locations from a well-known place, such as your house, are associated with the to-be-remembered items. Although specific mnemonic techniques are limited in their usefulness, the basic ideas behind them (utilizing imagery, forming meaningful associations, and using consistent encoding and retrieval strategies) are of broad value for improving memory performance.

Vu et al. (2007) examined the effectiveness of a "first-letter" mnemonic technique to help users relate individual characters of a password to a structured sentence in order to aid recall at a later time. In one condition, Vu et al. had users generate a sentence and take the first letter of each word in the sentence to form a password; in another condition, users generated a sentence that also included a number and special character embedded into the sentence and resulting password. Passwords generated using the first-letter technique were more memorable when users did not have to embed a digit and special character into the sentence, but were more secure (i.e., more resistant to cracking) when the sentence and resulting password included the digit and special character. Thus, when it comes to memory and security of computer passwords, there seems to be a trade-off between memorability and security.

Two additional factors have shown recently to benefit retrieval from memory. The first is repeated testing of items (Karpicke and Roediger 2008). The retrieval practice engendered by testing seems to be far more beneficial than additional studying of the items. The second is the relation of the to-be-remembered items to adaptive function (Nairne and Pandeirad 2010). Several studies have found evidence that survival-related words are retained better than ones that are not related to that adaptive function. From results like these, Nairne and Pandeirada have concluded, "to maximize retention in basic and applied settings it is useful to develop encoding techniques that are congruent with the natural design of memory systems" (p. 381).

2.6 ATTENTION IN INFORMATION PROCESSING

Attention is increased awareness directed at a particular event or action to select it for increased processing. This processing may result in enhanced understanding of the event, improved performance of an action, or better memory for the event. Attention allows us to filter out unnecessary information so that we can focus on a particular aspect that is relevant to our goals. Several significant information-processing models of attention have been proposed.

2.6.1 Models of Attention

In an influential study, Cherry (1953) presented different messages to each ear through headphones. Subjects were to repeat aloud one of the two messages while ignoring the other. When subsequently asked questions about the two messages, subjects were able to accurately describe the message to which they were attending but could not describe anything except physical characteristics, such as gender of the speaker, about the unattended message.

To account for such findings, Broadbent (1958) developed the filter theory, which assumes that the nervous system acts as a single-channel processor. According to filter theory, information is received in a preattentive temporary store and then is selectively filtered, based on physical features such as spatial location, to allow only one input to access the channel. Broadbent's filter theory implies that the meaning of unattended messages is not identified, but later studies showed that the unattended message could be processed beyond the physical level, in at least some cases (Treisman 1964).

To accommodate the finding that meaning of an unattended message can influence performance, Treisman (1964) reformulated filter theory into what is called the filter-attenuation theory. According to attenuation theory, early selection by filtering still precedes stimulus identification, but the filter only attenuates the information on unattended channels. This attenuated signal may be sufficient to allow identification if the stimulus is one with a low-identification threshold, such as a person's name or an expected event. Deutsch and Deutsch (1963) proposed that unattended stimuli are always identified and the bottleneck occurs in later processing, a view called late-selection theory. The difference between attenuation theory and late-selection theory is that the latter assumes that meaning is fully analyzed, whereas the former does not.

Lavie et al. (2004) have proposed a load theory of attention, which they claim "resolves the long-standing early versus late selection debate" (p. 339). Specifically, the load theory includes two selective attention mechanisms, a perceptual selection mechanism and a cognitive control mechanism. When perceptual load is high (i.e., great demands are placed on the perceptual system), the perceptual mechanism excludes irrelevant stimuli from being processed. When memory load is high, it is not possible to suppress irrelevant information at a cognitive level. In support of load theory, Lavie et al. showed that interference from distracting stimuli is reduced under conditions of high perceptual load but increased under conditions of high working memory load.

In divided attention tasks, a person must attend to multiple sources of information simultaneously. Kahneman (1973) proposed a unitary resource model that views attention as a single resource that can be divided up among different tasks in different amounts, based on task demands and voluntary allocation strategies. Unitary resource models provided the impetus for dual-task methodologies, such as performance operating characteristics, and mental workload analyses that are used widely in HCI (Eberts 1994). The expectation is that multiple tasks should produce interference when their resource demands exceed the supply that is available.

Many studies have shown that it is easier to perform two tasks together when they use different stimulus or response modalities than when they use the same modalities. Performance is also better when one task is verbal and the other visuospatial than when they are the same type. These result patterns provide the basis for multiple resource models of attention such as that of Wickens (1984). According to multiple resource models, different attentional resources exist for different sensory-motor modalities and coding domains. Multiple resource theory captures the fact that multiple-task performance typically is better when the tasks use different input–output modes than when they use the same modes. However, it is often criticized as being too flexible because new resources can be proposed arbitrarily to fit any finding of specificity of interference (Navon 1984).

A widely used metaphor for visual attention is that of a spotlight that is presumed to direct attention to everything in its field (Posner and Cohen 1984). Direction of attention is not necessarily the same as the direction of gaze because the attentional spotlight can be directed independently of fixation. Studies show that when a location is cued as likely to contain a target stimulus, but then a probe stimulus is presented at another location, a spatial gradient surrounds the attended location such that items nearer to the focus of attention are processed more efficiently than those farther away from it (Yantis 2000). The movement of the attentional spotlight to a location can be triggered by two types of cues: exogenous and endogenous. An exogenous cue is an external event such as the abrupt onset of a stimulus at a peripheral location that involuntarily draws the attentional spotlight to its location. Exogenous cues produce rapid performance benefits, which dissipate quickly, for stimuli presented at the cued location. This is followed by a period in which performance is worse for stimuli at the cued location than for ones presented at the uncued location, a phenomenon called inhibition of return (Posner and Cohen 1984). An endogenous cue is typically a symbol such as a central arrowhead that must be identified before a voluntary shift in attention to the designated location can be made. The performance benefits for endogenous cues take longer to develop and are sustained for a longer period of time when the cues are relevant, indicating that their benefits are due to conscious control of the attentional spotlight (Klein and Shore 2000).

Attentional focus is needed to detect change, and once attention is allocated to the processing of an event, there is a period in which it cannot be allocated to the processing of another event. Change blindness is the inability to detect sometimes large changes in a visual display or scene (Simons and Ambinder 2005). It has been demonstrated in the flicker task, in which one scene alternates with another and the presence versus absence of a distinctive feature such as an aircraft engine or building is not detected. Change blindness also occurs in natural settings when attention is diverted momentarily. A closely related phenomenon is that of the attentional blink (Martens and Wyble 2010). In this paradigm, there is rapid presentation of a sequence of displays of visual stimuli. When a target stimulus is detected in one display, the probability of detecting a second target stimulus presented within the next several displays is reduced dramatically. Martens and Wyble attribute the attentional blink to a deficit in consolidation of the second target into a conceptual working-memory representation due to processing capacity being devoted to consolidation of the first target and being unavailable for processing of the second. They note that this limitation may be linked to a mechanism of attentional control.

In a visual search task, subjects are to detect whether a target is present among distractors. Treisman and Gelade (1980) developed feature integration theory to explain the results from visual search studies. When the target is distinguished from the distractors by a basic feature such as color (feature search), RT and error rate often show little increase as the number of distractors increases. However, when two or more features must be combined to distinguish the target from distractors (conjunctive search), RT and error rate typically increase sharply as the number of distractors increases. To account for these results, feature integration theory assumes that basic features of stimuli are encoded into feature maps in parallel across the visual field at a preattentive stage. Feature search can be based on this preattentive stage because a “target-present” response requires only detection of the feature. The second stage involves focusing attention on a specific location and combining features that occupy the location into objects. Attention is required for conjunctive search because responses cannot be based on detection of a single feature. According to feature integration theory, performance in conjunctive search tasks decreases as the number of distractors increases because attention must be moved sequentially across the search field until a target is detected or all items present have been searched. Feature integration theory served to generate a large amount of research on visual search that showed, as typically the case, that the situation is not as simple as depicted by the theory. This has resulted in modifications of the theory, as well as alternative theories. For example, Wolfe’s (2007) Guided Search Theory maintains the distinction between an initial stage of feature maps and a second stage of attentional binding, but assumes that the second stage is guided by the initial feature analysis.

In HCI, a common visual search task involves locating menu items. When users know exactly what option to search for, identity matching can be used, in which users search the display for the menu name that they want to find. Perlman (1984) suggested that when identity search is used, the menu options should be displayed in alphabetical order to facilitate search. When users do not know where an option is included within a main list of menus, inclusion matching is used.

The users must decide within which group the specific option would be categorized and then search the list of items for that group. With inclusion matching, search times may be longer for items that can be classified in more than one of the main groupings or when the items are less well-known examples of a main grouping (Somberg and Picardi 1983). Equivalence search occurs when the users know what option to select, but does not know how that option is labeled. McDonald, Stone, and Liebelt (1983) showed that alphabetical and categorical organizations yield shorter search times than randomized organization for equivalence search. Search can also be affected by the breadth versus depth of the menu design. Lee and MacGregor (1985) showed that deep hierarchies are preferred over broad ones. However, more recently, Tullis, Tranquada, and Siegel (2011) suggested that, for complex or ambiguous situations, there is a benefit for broad menu designs because they facilitate comparison between categories. The main point is that when structuring menus, designers must consider the type of search in which the user would most likely be engaged.

The role of attention in response selection has been investigated extensively using the psychological refractory period (PRP) paradigm (Pashler 1998). In the PRP paradigm, a pair of choice-reaction tasks must be performed, and the stimulus onset asynchrony (SOA) of the second stimulus is presented at different intervals. RT for Task 2 is slowed at short SOAs, and this phenomenon is called the PRP effect. The experimental results have been interpreted with what is called locus of slack logic (Schweickert 1978), which is an extension of additive factors logic to dual-task performance. The basic idea is that if a Task 2 variable has its effect prior to a bottleneck, that variable will have an underadditive interaction with SOA. This underadditivity occurs because, at short SOAs, the slack period during which postbottleneck processing cannot begin can be used for continued processing for the more difficult condition. If a Task 2 variable has its effect after the bottleneck, the effect will be additive with SOA.

The most widely accepted account of the PRP effect is the response-selection bottleneck model (Pashler 1998). The primary evidence for this model is that perceptual variables typically have underadditive interactions with SOA, implying that their effects are before the bottleneck. In contrast, postperceptual variables typically have additive effects with SOA, implying that their effects are after the bottleneck. There has been dispute as to whether there is also a bottleneck at the later stage of response initiation (De Jong 1993), whether the response-selection bottleneck is better characterized as a parallel processor of limited capacity that divides resources among to-be-performed tasks (Tombu and Jolicœur 2005), and whether the apparent response-selection bottleneck is structural or simply a strategy adopted by subjects to comply with task instructions (Meyer and Kieras 1997). This latter approach is consistent with an emphasis on the executive functions of attention in the coordination and control of cognitive processes (Monsell and Driver 2000).

2.6.2 Automaticity and Practice

Attention demands are high when a person first performs a new task. However, these demands decrease and performance improves as the task is practiced. Because the quality of performance and attentional requirements change substantially as a function of practice, it is customary to describe performance as progressing from an initial cognitively demanding phase to a phase in which processing is automatic (Anderson 1982; Fitts and Posner 1967).

With the largest benefits occurring early in practice, the time to perform virtually any task from choice RT to solving geometry problems decreases with practice. Newell and Rosenbloom (1981) proposed a power function to describe the changes in RT with practice:

$$\mathrm{RT} = BN^{-\alpha} \tag{2.3}$$

where N is the number of practice trials, B is RT on the first trial, and α is the learning rate. Although the power function has become widely accepted as a law that describes the changes in RT, Heathcote, Brown, and Mewhort (2000) indicated that it does not fit the functions for individual performers adequately. They showed that exponential functions provided better fits than power functions to 40 individual data sets, and proposed a new exponential law of practice. The defining characteristic of the exponential function is that the relative learning rate is a constant at all levels of practice, whereas, for the power function, the relative learning rate is a hyperbolically decreasing function of practice trials.

2.7 PROBLEM SOLVING AND DECISION MAKING

Beginning with the work of Newell and Simon (1972), it has been customary to analyze problem solving in terms of a problem space. The problem space consists of the following: (1) an initial state, (2) a goal state that is to be achieved, (3) operators for transforming the problem from the initial state to the goal state in a sequence of steps, and (4) constraints on application of the operators that must be satisfied. The problem-solving process itself is conceived of as a search for a path that connects the initial and goal states.

Because the size of a problem space increases exponentially with the complexity of the problem, most problem spaces are well beyond the capacity of STM. Consequently, for problem solving to be effective, search must be constrained to a limited number of possible solutions. A common way to constrain search is through the use of heuristics. For example, people often use a means-ends heuristic for which at each step, an operator is chosen that will move the current state closer to the goal state (Atwood and Polson 1976). Such heuristics are called weak methods because they do not require much knowledge about the exact problem domain. Strong methods, such as those used by experts, rely on prior domain-specific knowledge and do not require much search because they are based on established principles applicable only to certain tasks.

The problem space must be an appropriate representation of the problem, if the problem is to be solved. One important method for obtaining an appropriate problem space is to use analogy or metaphor. Analogy enables a shift from a problem space that is inadequate to one that may allow the goal state to be reached. There are several steps in using analogies (Holland et al. 1986), including detecting similarity between source and target problems, and mapping the corresponding elements of the problems. Humans are good at mapping the problems, but poor at detecting that one problem is an analog of another. An implication for HCI is that potential analogs should be provided to users for situations in which they are confronted by novel problems.

The concept of mental model, which is closely related to that of the problem space, has become widely used in recent years (see Payne, this volume). The general idea of mental models with respect to HCI is that as the user interacts with the computer, she or he receives feedback from the system that allows him/her to develop a representation of how the system is functioning for a given task. The mental model incorporates the goals of the user, the actions taken to complete the goals, and expectations of the system's output in response to the actions. A designer can increase the usability of an interface by using metaphors that allow transfer of an appropriate mental model (e.g., the desktop metaphor), designing the interface to be consistent with other interfaces with which the user is familiar (e.g., the standard web interface), and conveying the system's functions to the user in a clear and accurate manner. Feedback to the user is perhaps the most effective way to communicate information to the user and can be used to guide the user's mental model about the system.

Humans often have to make choices for situations in which the outcome depends on events that are outside of their control. According to expected utility theory, a normative theory of decision making under uncertainty, the decision maker should determine the expected utility of a choice by multiplying the subjective utility of each outcome by the outcome's probability and summing the resulting values (Hastie and Dawes 2010). The expected utility should be computed for each choice, and the optimal decision is the choice with the highest expected utility. It should be clear from this description that for all but the simplest of problems, a human decision maker cannot operate in this manner. To do so would require attending to multiple cues that exceed attentional capacity, accurate estimates of probabilities of various events, and maintenance of, and operation on, large amounts of information that exceeds STM capacity.

Research of Kahneman and Tversky (2000) and others has shown that what people do when the outcome associated with a choice is uncertain is to rely heavily on decision-making heuristics. These heuristics include representativeness, availability, and anchoring. The representativeness heuristic is that the probability of an instance being a member of a particular category is judged on the basis of how representative the instance is of the category. The major limitation of the representativeness heuristic is that it ignores base rate probabilities for the respective categories. The availability heuristic involves determining the probability of an event based on the ease with which instances of the event can be retrieved. The limitation is that availability is affected not only by relative frequency but also by other factors. The anchoring heuristic involves making a judgment regarding probabilities of alternative states based on initial information, and then adjusting these probabilities from this initial "anchor" as additional information is received. The limitation of anchoring is that the initial judgment can produce a bias for the probabilities. Although heuristics are useful, they may not always lead to the most favorable decision. Consequently, designers need to make sure that the choice desired for the user in a particular situation is one that is consistent with the user's heuristic biases.

2.8 SUMMARY AND CONCLUSION

The methods, theories, and models in human information processing are currently well developed. The knowledge in this area, of which we are only able to describe at a surface level in this chapter, is relevant to a wide range of concerns in HCI, from visual display design to representation and communication of knowledge. For HCI to be effective, the interaction must be made compatible with the human information-processing capabilities. Cognitive architectures that incorporate many of the facts about human information processing have been developed that can be applied to HCI. The Model Human Processor of Card, Moran, and Newell (1983) is the most widely known, but other more recent architectures, including the adaptive control of thought model of Anderson and colleagues (Anderson, Matessa, and Lebiere 1997), the State, Operator, and Result (SOAR) Model of Newell and colleagues (Howes and Young 1997), and the EPIC Model of Kieras and Meyer (1997) have considerable utility for the field, as demonstrated in Chapter 5 of this volume.

The human information-processing approach emphasizes laboratory research in which fundamental cognitive processes and principles thought to be of broad generalizability are established. Although this approach has been highly successful in many respects, some researchers think that more emphasis should be placed on real-world behavior in natural environments. Alternative approaches to perception, cognition, and action with such emphasis include the following. The ecological approach associated with Gibson (1979) places emphasis on analyzing the perceptual information that is available in the optic array and the dynamics of this information as the individual interacts with the environment. The cybernetic view, that cognition emerges as a consequence of motor control over sensory feedback, stresses self-regulated control of perception and cognition (Smith and Henning 2005). The situated cognition approach focuses on the need to understand behavior in specific contexts in which, for example, a computer application will be used (Kiekel and Cooke 2011). A recently popular approach is that of embodied cognition, according to which knowledge is acquired and processed through interactions of the body with the environment (e.g., Sherman, Gangi, and White 2010). One common feature of these alternative accounts is an emphasis on the relation among perception, cognition, and action. We agree that, in certain areas of information-processing research, action has been viewed as a final stage that does not influence the prior stages of perception and cognition. However, in other areas, such as that of human performance, action has been emphasized since the earliest applications of the information-processing approach (Fitts and Posner 1967; since 1975, one division of the *Journal of Experimental Psychology* has been subtitled *Human Perception and Performance*). From our perspective, information-processing analyses and models will continue to be useful tools for understanding and predicting human behavior both in general and in HCI in particular.

REFERENCES

Anderson, J. R. 1982. Acquisition of cognitive skill. *Psychol Rev* 89:369–406.

Anderson, J. R., D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Qin. 2004. An integrated theory of the mind. *Psychol Rev* 111:1036–60.

Anderson, J. R., J. M. Fincham, and S. Douglass. 1999. Practice and retention: A unifying analysis. *J Exp Psychol Learn Mem Cogn* 25:1120–36.

Anderson, J. R., M. Matessa, and C. Lebiere. 1997. ACT-R: A theory of higher level cognition and its relation to visual attention. *Hum Comput Interact* 12:439–62.

Atwood, M. E., and P. G. Polson. 1976. A process model for water jug problems. *Cogn Psychol* 8:191–216.

Baddeley, A. D. 2000. The episodic buffer: A new component of working memory? *Trends Cogn Sci* 4:421.

Baddeley, A. D. 2003. Working memory and language: An overview. *J Commun Disord* 36:189–208.

Baddeley, A. D., and G. J. Hitch. 1974. Working memory. In *The Psychology of Learning and Motivation*, ed. G. H. Bower, vol. 8, 47–89. New York: Academic Press.

Baddeley, A. D., N. Thomson, and M. Buchanan. 1975. Word length and the structure of shortterm memory. *J Verbal Learn Behav* 14:575–89.

Bahrick, H. P., and L. K. Hall. 2005. The importance of retrieval failures to long-term retention: A metacognitive explanation of the spacing effect. *J Mem Lang* 52:566–77.

Bertelson, P. 1967. The time course of preparation. *Q J Exp Psychol* 19:272–79.

Boldini, A., R. Russo, and S. E. Avons. 2004. One process is not enough! A speed-accuracy tradeoff study of recognition memory. *Psychon Bull Rev* 11:353–61.

Bower, G. H., and E. R. Hilgard. 1981. *Theories of Learning*. 5th ed. Englewood Cliffs, NJ: Prentice Hall.

Broadbent, D. E. 1958. *Perception and Communication*. Oxford, UK: Pergamon Press.

Brown, J. (1958). Some tests of the decay theory pf immediate memory. *Q J Exp Psychol* 10:12–21.

Busemeyer, J. R., and A. Diederich. 2010. *Cognitive Modeling*. Thousand Oaks, CA: Sage.

Card, S. K., W. K. English, and B. J. Burr. 1978. Evaluation of the mouse, rate-controlled isometrick joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* 21:601–13.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Erlbaum.

Chen, Z., and N. Cowan. 2009. Core verbal working-memory capacity: The limit in words retained without covert articulation. *Q J Exp Psychol* 62:1420–9.

Cherry, E. C. 1953. Some experiments on the recognition of speech, with one and with two ears. *J Acoust Soc Am* 25:975–9.

Craik, F. I. M., and R. S. Lockhart. 1972. Levels of processing: A framework for memory research. *J Verbal Learn Verbal Behav* 11:671–84.

Craik, F. I. M., and M. J. Watkins. 1973. The role of rehearsal in short-term memory. *J Verbal Learn Verbal Behav* 12:599–607.

De Jong, R. 1993. Multiple bottlenecks in overlapping task performance. *J Exp Psychol Hum Percept Perform* 19:965–80.

Deutsch, J. A., and D. Deutsch. 1963. Attention: Some theoretical considerations. *Psychol Rev* 70:80–90.

Donders, F. C. 1868/1969. On the speed of mental processes. In *Acta Psychologica, 30, Attention and Performance II*, ed. W. G. Koster, 412–31. Amsterdam, Netherlands: North-Holland.

Duncan, J. 1977. Response selection rules in spatial choice reaction tasks. In *Attention and performance VI*, ed. S. Dornic, 49–71. Hillsdale, NJ: Erlbaum.

Eberts, R. E. 1994. *User Interface Design*. Englewood Cliffs, NJ: Prentice Hall.

Eimer, M. 1998. The lateralized readiness potential as an on-line measure of central response activation processes. *Behav Res Methods Instrum Comput* 30:146–56.

Fitts, P. M. 1951. Engineering psychology and equipment design. In *Handbook of Experimental Psychology*, ed. S. S. Stevens, 1287–340. New York: Wiley.

Fitts, P. M. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *J Exp Psychol* 47:381–91.

Fitts, P. M., and R. L. Deininger. 1954. S-R compatibility: Correspondence among paired elements within stimulus and response codes. *J Exp Psychol* 48:483–92.

Fitts, P. M., and M. I. Posner. 1967. *Human Performance*. Belmont, CA: Brooks/Cole.

Fitts, P. M., and C. M. Seeger. 1953. S-R compatibility: Spatial characteristics of stimulus and response codes. *J Exp Psychol* 46:199–210.

Gibson, J. J. 1979. *The Ecological Approach to Visual Perception*. Boston, MA: Houghton Mifflin.

Gillan, D. J., K. Holden, S. Adam, M. Rudisill, and L. Magee. 1992. How should Fitts's law be applied to human-computer interaction? *Interact Comput* 4:291–313.

Godden, D. R., and A. D. Baddeley. 1975. Context-dependent memory in two natural environments: On land and underwater. *Br J Psychol* 66:325–31.

Gonzalez, C., F. J. Lerch, and C. Lebiere. 2003. Instance-based learning in real-time dynamic decision making. *Cogn Sci* 27:591–635.

Grobelny, J., W. Karwowski, and C. Drury. 2005. Usability of graphical icons in the design of human-computer interfaces. *Int J Hum Comput Interact* 18:167–82.

Guiard, Y., and M. Beaudouin-Lafon. 2004. Fitts's law 50 years later: Applications and contributions from human-computer interaction. *Int J Hum Comput Stud* 61:747–50.

Hastie, R., and R. M. Dawes. 2010. *Rational Choice in An Uncertain World: The Psychology of Judgment and Decision Making*. Thousand Oaks, CA: Sage.

Healy, A. F., E. L. Wohldmann, and L. E. Bourne Jr. 2005. The procedural reinstatement principle: Studies on training, retention, and transfer. In *Experimental Cognitive Psychology and Its Applications*, ed. A. F. Healy, 59–71. Washington, DC: American Psychological Association.

Heathcote, A., S. Brown, and D. J. K. Mewhort. 2000. The power law repealed: The case for an exponential law of practice. *Psychono Bull Rev* 7:185–207.

Hick, W. E. 1952. On the rate of gain of information. *Q J Exp Psychol* 4:11–26.

Hintzman, D. L. 1974. Theoretical implications of the spacing effect. In *Theories of Cognitive Psychology: The Loyola Symposium*, ed. R. L. Solso, 77–99. Hillsdale, NJ: Erlbaum.

Hoffman, R. R., and K. A. Deffenbacher. 1992. A brief history of applied cognitive psychology. *Appl Cogn Psychol* 6:1–48.

Holland, J. H., K. J. Holyoak, R. E. Nisbett, and P. R. Thagard. 1986. *Induction*. Cambridge, MA: MIT Press.

Hommel, B., and W. Prinz, eds. 1997. *Theoretical Issues in Stimulus-Response Compatibility*. Amsterdam, Netherlands: North-Holland.

Howes, A., and R. M. Young. 1997. The role of cognitive architecture in modeling the user: Soar's learning mechanism. *Hum Comput Interact* 12:311–43.

Huettel, S. A., A. W. Song, and G. McCarthy. 2004. *Functional Magnetic Resonance Imaging*. Sunderland, MA: Sinauer Associates.

Hyde, T. S., and J. J. Jenkins. 1973. Recall of words as a function of semantic, graphic, and syntactic orienting tasks. *J Verbal Learn Verbal Behav* 12:471–80.

Hyman, R. 1953. Stimulus information as a determinant of reaction time. *J Exp Psychol* 45:188–96.

Jacko, J. A., and K. G. Ward. 1996. Toward establishing a link between psychomotor task complexity and human information processing. In *19th International Conference on Computers and Industrial Engineering*, vol. 31, 533–6.

Jacoby, L. L., and M. Dallas. 1981. On the relationship between autobiographical memory and perceptual learning. *J Exp Psychol Gen* 110:306–40.

Kahneman, D. 1973. *Attention and Effort*. Englewood Cliffs, NJ: Prentice Hall.

Kahneman, D., and A. Tversky, eds. 2000. *Choices, Values, and Frames*. New York: Cambridge University Press.

Karpicke, J. D., and H. L. Roediger III. 2008. The critical importance of retrieval for learning. *Science* 319:966–8.

Kauhanen, L., P. Jylänki, J. Lehtonen, P. Rantanen, H. Alaranta, and M. Sams. 2007. EEG-based brain-computer interface for tetraplegics. *Compu Intell Neurosci* Volume 2007, Article ID 23864, 11 pages doi:10.1155/2007/23864.

Keppel, G., and B. J. Underwood. 1962. Proactive inhibition in short-term retention of single items. *J Verbal Learn Verbal Behav* 1:153–61.

Kiekel, P. A., and N. J. Cooke. 2011. Human factors aspects of team cognition. In *Handbook of Human Factors in Web Design*, 2nd ed., ed. K.-P. L. Vu and R. W. Proctor, 107–23. Boca Raton, FL: CRC Press.

Kieras, D. E., and D. E. Meyer. 1997. An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Hum Comput Interact* 12:391–438.

Klein, R. M., and D. I. Shore. 2000. Relation among modes of visual orienting. In *Control of Cognitive Processes: Attention and Performance XVIII*, ed. S. Monsell and J. Driver, 195–208. Cambridge, MA: MIT Press.

Kornblum, S., T. Hasbroucq, and A. Osman. 1990. Dimensional overlap: Cognitive basis for stimulus-response compatibility—A model and taxonomy. *Psychol Rev* 97:253–70.

Kunde, W. 2001. Response-effect compatibility in manual choice reaction tasks. *J Exp Psychol Hum Percept Perform* 27:387–94.

Lachman, R., J. L. Lachman, and E. C. Butterfield. 1979. *Cognitive Psychology and Information Processing: An Introduction*. Hillsdale, NJ: Erlbaum.

Lavie, N., A. Hirst, J. W. de Fockert, and E. Viding. 2004. Load theory of selective attention and cognitive control. *J Exp Psychol Gen* 133:339–54.

Lee, Y.-S. 2008. Levels-of-processing effects on conceptual automatic memory. *Eur J Cogn Psychol* 20:936–54.

Lee, E., and J. MacGregor. 1985. Minimizing user search time in menu retrieval systems. *Hum Factors* 27:157–62.

Lees, M. N., J. D. Cosman, J. D. Lee, M. Rizzo, and N. Fricke. 2010. Translating cognitive neuroscience to the driver's operational environment: A neuroergonomics approach. *Am J Psychol* 123:391–411.

Loftus, E. F., and J. C. Palmer. 1974. Reconstruction of automobile destruction: An example of the interaction between language and memory. *J Exp Psychol Hum Learn Mem* 4:19–41.

Lu, C.-H., and R. W. Proctor. 1995. The influence of irrelevant location information on performance: A review of the Simon effect and spatial Stroop effects. *Psychon Bull Rev* 2:174–207.

MacKenzie, I. S., and R. W. Soukoreff. 2002. Text entry for mobile computing: Models and methods, theory and practice. *Hum Comput Interact* 17:147–98.

Mackworth, N. H. 1948. The breakdown of vigilance during prolonged visual search. *Q J Exp Psychol* 1:6–21.

Macmillan, N. A., and C. D. Creelman. 2005. *Detection Theory: A User's Guide*. 2nd ed. Mahwah, NJ: Erlbaum.

Martens, S., and B. Wyble. 2010. The attentional blink: Past, present, and future of a blind spot in perceptual awareness. *Neurosci Biobehav Rev* 34:947–57.

McCarley, J. S. 2009. Effects of speed-accuracy instructions on oculomotor scanning and target recognition in a simulated baggage X-ray screening task. *Ergonomics* 52:325–33.

McClelland, J. L. 1979. On the time relations of mental processes: A framework for analyzing processes in cascade. *Psychol Rev* 88:375–407.

McDonald, J. E., J. D. Stone, and L. S. Liebelt. 1983. Searching for items in menus: The effects of organization and type of target. In *Proceedings of the Human Factors Society 27th Annual Meeting*, 289–338. Hillsdale, NJ: Erlbaum.

McGuffin, M. J., and R. Balakrishnan. 2005. Fitts's law and expanding targets: Experimental studies and designs for user interfaces. *ACM Trans Comput Hum Interact* 12:388–422.

Melara, R. D., H. Wang, K.-P. L. Vu, and R. W. Proctor. 2008. Attentional origins of the Simon effect: Behavioral and electrophysiological evidence. *Brain Res* 1215:147–59.

Meyer, D. E., R. A. Abrams, S. Kornblum, C. E. Wright, and J. E. K. Smith. 1988. Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychol Rev* 86:340–70.

Meyer, D. E., and D. E. Kieras. 1997. A computational theory of executive cognitive processes and multiple-task performance: Part 2. Accounts of psychological refractory-period phenomena. *Psychol Rev* 104:749–91.

Miller, G. A. 1956. The magical number seven plus or minus two: Some limits on our capacity for processing information. *Psychol Rev* 63:81–97.

Miller, J. 1988. Discrete and continuous models of human information processing: Theoretical distinctions and empirical results. *Acta Psychol* 67:191–257.

Monsell, S., and J. Driver, eds. 2000. *Control of Cognitive Processes: Attention and Performance XVIII*. Cambridge, MA: MIT Press.

Morris, C. D., J. D. Bransford, and J. J. Franks. 1977. Levels of processing versus transfer appropriate processing. *J Verbal Learn Verbal Behav* 16:519–33.

Nairne, J. S. 2003. Sensory and working memory. In *Experimental Psychology*. vol. 4 of *Handbook of Psychology*, ed. A. F. Healy and R. W. Proctor. Editor-in-Chief: I. B. Weiner. Hoboken, NJ: Wiley.

Nairne, J. S., and Pandeirada, J. N. S. 2010. Adaptive memory: Nature's criterion and the functionalist agenda. *Am J* Psychol 123:381–90.

Navon, D. 1984. Resources—a theoretical soup stone? *Psychol Rev* 91:216–34.

Newell, A., and P. S. Rosenbloom. 1981. Mechanisms of skill acquisition and the law of practice. In *Cognitive Skills and Their Acquisition*, ed. J. R. Anderson, 1–55. Hillsdale, NJ: Erlbaum.

Newell, A., and H. A. Simon. 1972. *Human Problem Solving*. Englewood Cliffs, NJ: Prentice Hall.

Oulasvirta, A., L. Kärkkäinen, and J. Laarni. 2005. Expectations and memory in link search. *Comput Hum Behav* 21:773–89.

Oulasvirta, A., and P. Saariluoma. 2004. Long-term working memory and interrupting messages in human-computer interaction. *Behav Inf Technol* 23:53–64.

Pachella, R. G. 1974. The interpretation of reaction time in information-processing research. In *Human Information Processing: Tutorials in Performance and Cognition*, ed. B. H. Kantowitz, 41–82. Hillsdale, NJ: Erlbaum.

Parasuraman, R. 2003. Neuroergonomics: Research and practice. *Theor Issues Ergon Sci* 4:5–20.

Parasuraman, R., and D. R. Davies. 1977. A taxonomic analysis of vigilance performance. In *Vigilance: Theory, Operational Performance, and Physiological Correlates*, ed. R. R. Mackie, 559–74. New York: Plemum.

Park, H., and M. D. Rugg. 2008. The relationship between study processing and the effects of cue congruency at retrieval: fMRI support for transfer appropriate processing. *Cereb Cortex* 18:868–75.

Parsons, M. W., D. L. Harrington, and S. M. Rao. 2005. Distinct neural systems underlie learning visuomotor and spatial representations of motor skills. *Hum Brain Mapp* 24:229–47.

Pashler, H. 1998. *The Psychology of Attention*. Cambridge, MA: MIT Press.

Payne, S. J. 1995. Naïve judgments of stimulus-response compatibility. *Hum Factors* 37:495–506.

Percival, L. C., and T. K. Noonan. 1987. Computer network operation: Applicability of the vigilance paradigm to key tasks. *Hum Factors* 29:685–94.

Perlman, G. 1984. Making the right choices with menus. *Proceedings of INTERACT '84*, 291–5. London, UK: IFIP.

Peterson, L. R., and M. J. Peterson. 1959. Short-term retention of individual verbal items. *J Exp Psychol* 58:193–8.

Posner, M. I. 1986. Overview. In *Handbook of Perception and Human Performance*. vol. 2 of *Cognitive Processes and Performance*, ed. K. R. Boff, L. Kaufman, and J. P. Thomas, V3–10. New York: Wiley.

Posner, M. I., and Y. Cohen. 1984. Components of visual orienting. In *Attention and Performance X*, ed. H. Bouma, and D. G. Bouwhuis, 531–56. Hillsdale, NJ: Erlbaum.

Proctor, R. W., and Y. S. Cho. 2006. Polarity correspondence: A general principle for performance of speeded binary classification tasks. *Psychol Bull* 132:416–42.

Proctor, R. W., and T. Van Zandt. 2008. *Human Factors in Simple and Complex Systems*. 2nd ed. Boca Raton, FL: CRC Press.

Proctor, R. W., and K.-P. L. Vu. 2006. *Stimulus-Response Compatibility: Data, Theory and Application*. Boca Raton, FL: CRC Press.

Proctor, R. W., and K.-P. L. Vu. 2010. Cumulative knowledge and progress in human factors. *Annu Rev Psychol* 61:623–51.

Proctor, R. W., K.-P. L. Vu, and D. F. Pick. 2005. Aging and response selection in spatial choice tasks. *Hum Factors* 47:250–70.

Ratcliff, R., and P. L. Smith. 2004. A comparison of sequential sampling models for two-choice reaction time. *Psychol Rev* 111:333–67.

Reeve, T. G., and R. W. Proctor. 1990. The salient features coding principle for spatial- and symbolic-compatibility effects. In *Stimulus-Response Compatibility: An Integrated Perspective*, ed. R. W. Proctor and T. G. Reeve, 163–80. Amsterdam, Netherlands: North-Holland.

Roediger III, H. L., and K. B. McDermott. 1995. Creating false memories: Remembering words not presented in lists. *J Exp Psychol Learn Mem Cogn* 21:803–14.

Roscoe, S. 2011. Historical overview of human factors and ergonomics. In *Handbook of Human Factors in Web Design*, 2nd ed., ed. K.-P. L. Vu and R. W. Proctor, 3–12. Boca Raton: CRC Press.

Rotello, C. M., and N. A. Macmillan. 2006. Remember–know models as decision strategies in two experimental paradigms. *J Mem Lang* 55:479–94.

Rubichi, S., K.-P. L. Vu, R. Nicoletti, and R. W. Proctor. 2006. Spatial coding in two dimensions. *Psychon Bull Rev* 13:201–16.

Rugg, M. D., and M. G. H. Coles, eds. 1995. *Electrophysiology of Mind: Event-Related Brain Potentials and Cognition*. Oxford, England UK: Oxford University Press.

Salthouse, T. A. 2005. From description to explanation in cognitive aging. In *Cognition and Intelligence: Identifying the Mechanisms of the Mind*, ed. R. J. Sternberg and J. E. Pretz, 288–305. Cambridge: Cambridge University Press.

Sanders, A. F. 1998. *Elements of Human Performance*. Mahwah, NJ: Erlbaum.

Sartori, G., and C. Umiltà. 2000. How to avoid the fallacies of cognitive subtraction in brain imaging. *Brain Lang* 74:191–212.

Schweickert, R. 1978. A critical path generalization of the additive factor method: Analysis of a Stroop task. *J Math Psychol* 18:105–39.

Schweickert, R., and B. Boruff. 1986. Short-term memory capacity: Magic number or magic spell? *J Exp Psychol Learn Mem Cogn* 12:419–25.

Shaffer, L. H. 1965. Choice reaction with variable S-R mapping. *J Exp Psychol* 70:284–8.

Shepard, R. N., and J. Metzler. 1971. Mental rotation of three-dimensional objects. *Science* 171:701–3.

Sherman, D. K., C. Gangi, and M. L. White. 2010. Embodied cognition and health persuasion: Facilitating intention–behavior consistency via motor manipulations. *J Exp Soc Psychol* 46:461–4.

Simon, J. R. 1990. The effects of an irrelevant directional cue on human information processing. In *Stimulus-Response Compatibility: An Integrated Perspective*, ed. R. W. Proctor and T. G. Reeve, 31–86. Amsterdam: North-Holland.

Simons, D. J., and M. S. Ambinder. 2005. Change blindness: Theory and consequences. *Curr Dir Psychol Sci* 14:44–8.

Slamecka, N. J., and P. Graf. 1978. The generation effect: Delineation of a phenomenon. *J Exp Psychol Hum Learn Mem* 4:592–604.

Smith, T. J., and R. A. Henning. 2005. Cybernetics of augmented cognition as an alternative to information processing. In *Proceedings of HCI International 2005*. Mahwah, NJ: Erlbaum.

Somberg, B. L., and M. C. Picardi. 1983. Locus of information familiarity effect in search of computer menus. In *Proceedings of the Human Factors Society 27th Annual Meeting*, 826–30. Santa Monica, CA: Human Factors Society.

Sternberg, S. 1969. The discovery of processing stages: Extensions of Donders' method. In *Attention and Performance II Acta Psychologica*, ed. W. G. Koster, vol. 30, 276–315. North Holland: Amsterdam.

Sternberg, S., S. Monsell, R. L. Knoll, and C. E. Wright. 1978. The latency and duration of rapid movement sequences. In *Information Processing in Motor Control and Learning*, ed. G. E. Stelmach, 117–52. New York: Academic Press.

Stevenson, R. A., S. Kim, and T. W. James. 2009. An additive-factors design to disambiguate neuronal and areal convergence: Measuring multisensory interactions between audio, visual, and haptic sensory streams using fMRI. *Exp Brain Res* 198:183–94.

Surprenant, A. M., and I. Neath. 2009. *Principles of Memory*. Philadelphis, PA: Psychology Press.

Talmi, D., C. L. Grady, Y. Goshen-Gottstein, and M. Moscovitch. 2005. Neuroimaging the serial position curve: A test of single-store versus dual-store models. *Psychol Sci* 16:716–23.

Teichner, W. H., and M. J. Krebs. 1974. Laws of visual choice reaction time. *Psychol Rev* 81:75–98.

Tlauka, M. 2004. Display-control compatibility: The relationship between performance and judgments of performance. *Ergonomics* 47:281–95.

Tombu, M., and P. Jolicœur. 2005. Testing the predictions of the central capacity sharing model. *J Exp Psychol Hum Percept Perform* 31:790–802.

Treisman, A. M. 1964. Selective attention in man. *Br Med Bull* 20:12–6.

Treisman, A. M., and G. Gelade. 1980. A feature-integration theory of attention. *Cogn Psychol* 12:97–136.

Trimmel, M., and R. Huber. 1998. After-effects of human-computer interaction indicated by P300 of the event-related brain potential. *Ergonomics* 41:649–55.

Tullis, T. S., F. J. Tranquada, and M. J. Siegel. 2011. Presentation of information. In *Handbook of Human Factors in Web Design*, 2nd ed., ed. K.-P. L. Vu and R. W. Proctor, 153–89. Boca Raton, FL: CRC Press.

Usher, M., Z. Olami, and J. L. McClelland. 2002. Hick's law in a stochastic race model with speed-accuracy tradeoff. *J Math Psychol* 46:704–15.

van de Laar, M. C., W. P. M. van den Wildenberg, G. J. M. van Boxtel, and M. W. van der Molen. 2010. Processing of global and selective stop signals application of Donders' subtraction method to stop-signal task performance. *Exp Psychol* 57:149–59.

Van Zandt, T., H. Colonius, and R. W. Proctor. 2000. A comparison of two response time models applied to perceptual matching. *Psychon Bull Rev* 7:208–56.

Verhaeghen, P., and A. Marcoen. 1996. On the mechanisms of plasticity in young and older adults after instruction in the method of loci: Evidence for an amplification model. *Psychol Aging* 11:164–78.

Vu, K.-P. L., and R. W. Proctor. 2003. Naïve and experienced judgments of stimulus-response compatibility: Implications for interface design. *Ergonomics* 46:169–87.

Vu, K.-P. L., and R. W. Proctor. 2004. Mixing compatible and incompatible mappings: Elimination, reduction, and enhancement of spatial compatibility effects. *Q J Exp Psychol* 57A:539–56.

Vu, K.-P. L., and R. W. Proctor. 2008. Age differences in response selection for pure and mixed stimulus-response mappings and tasks. *Acta Psychol* 129:49–60.

Vu, K.-P. L., R. W. Proctor, A. Bhargav-Spanzel, B. -L. Tai, J. Cook, and E. E. Schultz. 2007. Improving password security and memorability to protect personal and organizational information. *Int J Hum Comput Stud* 65:744–57.

Walker, N., J. B. Smelcer, and E. Nilsen. 1991. Optimizing speed and accuracy of menu selection: A comparison of walking and pull-down menus. *Int J Man Mach Stud* 35:871–90.

Wang, H., and R. W. Proctor. 1996. Stimulus-response compatibility as a function of stimulus code and response modality. *J Exp Psychol Hum Percept Perform* 22:1201–17.

Ward, J. 2010. *The Student's Guide to Cognitive Neuroscience*. 2nd ed. New York: Psychology Press.

Warm, J. S., R. Parasuraman, and G. Matthews. 2008. Vigilance requires hard mental work and is stressful. *Hum Factors* 50:433–41.

Wickelgren, W. A. 1977. Speed-accuracy tradeoff and information processing dynamics. *Acta Psychol* 41:67–85.

Wickens, D. D. 1970. Encoding categories of words: An empirical approach to meaning. *Psychol Rev* 77:1–15.

Wickens, C. D. 1984. Processing resources in attention. In *Varieties of Attention*, ed. R. Parasuraman and D. R. Daives, 63–102. San Diego, CA: Academic Press.

Wickens, C. D., S. Gordon, Y. Liu, and J. Lee. 2003. *An Introduction to Human Factors Engineering*. 2nd ed. New York: Prentice Hall.

Wolfe, J. M. 2007. Guided Search 4.0: Current progress with a model of visual search. In *Integrated Models of Cognitive Systems*, ed. W. D. Gray, 99–119. New York: Oxford University Press.

Wood, L. E., and J. D. Pratt. 1987. Pegword mnemonic as an aid to memory in the elderly: A comparison of four age groups. *Educ Gerontol* 13:325–39.

Yantis, S. 2000. Goal-directed and stimulus-driven determinants of attentional control. In *Control of Cognitive Processes: Attention and Performance XVIII*, ed. S. Monsell and J. Driver, 195–208. Cambridge, MA: MIT Press.

Young, P. M., B. A. Clegg, and C. A. P. Smith. 2004. Dynamic models of augmented cognition. *Int J Hum Comput Interact* 17:259–73.

3 Mental Models in Human–Computer Interaction

Stephen J. Payne

CONTENTS

The plan for this chapter is as follows. It begins by reviewing and discussing the term "mental models" as it has been used in the literature on human–computer interaction (HCI), and in the neighboring disciplines of cognitive psychology where it was first coined. There is little consensus on what exactly is and is not a mental model, and yet it is too widely used for any posthoc attempt at a narrower definition to somehow cleanse the field. In consequence, I characterize several layers of theoretical commitment that the term may embrace, following an earlier discussion (Payne 2003). To illustrate the argument, several classic and more recent studies from the HCI literature will be reviewed, with pointers to others. This first part of the chapter is based on material published in Payne (2003).

In cognitive psychology, mental models have major currency in two sub-disciplines—text comprehension and reasoning, although in the former they more often currently go by the name "situation models." Discussion in the latter focuses on quite refined theoretical disputes that currently have little relevance for HCI. The work on text comprehension, however, is germane. With the advent of the web, the comprehension of text of various kinds has become a dominant mode of HCI, with important design issues for websites, digital libraries, and so on. Interaction with text is in some ways a paradigm for interaction with information. With these points in mind, the concept of mental models in text comprehension will be discussed, with a particular eye to the issues that HCI accentuates, such as understanding multiple texts.

Two of the major practical questions raised by mental models are (1) How are they acquired? and (2) How can their acquisition be supported by instruction? The third section of this chapter will discuss two angles on these questions in HCI: first, the use of interactive computation and multimedia as an instructional method; second, the important tension between exploration and instruction, first systematically discussed in the HCI literature by Carroll's (1990) work on minimalism.

Finally, the paper will review some recent work on the importance of mental models for understanding aspects of collaborative teamwork. This area suggests that a relatively expansive view of human knowledge representations may be necessary for progress in HCI.

Throughout the chapter, a particular approach is taken to review: to choose one or two key studies and report them in some detail. I hope that this will allow some of the empirical methodologies and the rich variation in these to be conveyed. The chosen studies will be accompanied by some further references to the literature, but there are too many subtopics reviewed to aim for completeness.

3.1 WHAT IS A MENTAL MODEL?

The user's mental model of the device is one of the more widely discussed theoretical constructs in HCI. Alongside wide-ranging research literature, even commercial style guides have appealed to mental models for guidance (i.e., Mayhew 1992; Tognazzini 1992; Apple Human Interface Guidelines Apple Computer Inc. 1987).

Yet a casual inspection of the HCI literature reveals that mental models are used to label many different aspects of users' knowledge about the systems they use. Nevertheless, I propose that even this simple core construct—what users know and believe about the systems they use—is worth highlighting and promoting. It is more distinctive than it might first seem, especially in comparison with other cognitive-science approaches. Further, beyond the core idea there is a progression of stronger theoretical commitments that have been mobilized by the mental models label, each of which speaks to important issues in HCI research, if not yet in practice.

The fundamental idea is that the *contents* of people's knowledge, including their theories and beliefs, can be an important explanatory concept for understanding users' behavior in relation to systems. This idea may seem obvious and straightforward, but in fact it suggests research questions that go against the grain of most contemporary cognitive psychology, which has concerned itself much more with the general limits of the human-information-processing system, such as the constraints on attention, retrieval, and processing. Thus, cognitive psychology tends to focus on the *structure* of the mind, rather than its contents. (The major exception to the rule that cognitive psychology has been obsessed with architecture over content is the work on expertise, and even here, recent work has focused on explanations of extreme performance in terms of general independent variables such as "motivated practice," i.e., Ericsson, Krampe, and Tesch-Romer [1993], rather than epistemological analysis.)

Refocusing attention on mental content about particular domains is what made mental models a popular idea in the early 1980s, such as the papers in Gentner and Stevens (1983). For example, work on naïve physics (i.e., McCloskey 1983) attempts to explain people's reasoning about the physical world, not in terms of working memory limits or particular representations, but in terms of their beliefs about the world, such as the nature of their theories of mechanics or electricity, for example. This focus on people's knowledge, theories, and beliefs about particular domains transfers naturally to questions in HCI, where practical interest may focus on how users conceive the workings of a particular device, how their beliefs shape their interactive behavior, and what lessons may be drawn for design.

In this mold, consider a very simple study of my own (Payne 1991). Students were interviewed about ATMs. Following Collins and Gentner (1987) among others, "what if" questions were posed to uncover student's theories about the design and function of ATMs. For example, students were asked whether machines sometimes took longer to process their interactions; what information was stored on the plastic card; and what would happen if they "typed ahead" without waiting for the next machine prompt.

The interviews uncovered a wide variety in students' beliefs about the design of ATMs. For example, some assumed that the plastic card was written to as well as read from during transactions, and thus could encode the current balance of their account. Others assumed that the only information on the card was the user's personal identification number, allowing the machine to check the identity of the user (as it turns out, both these beliefs are incorrect). A conclusion from this simple observation is that users of machines are eager to form explanatory models and will readily go beyond available data to infer models that are consistent with their experiences. (One might wonder whether such explanations were not merely ad hoc, prompted during the interview: in fact some were, but explicit linguistic cues—such as "I've always thought"—strongly suggested that many were not.)

Another observation concerning students' "models" of ATMs was that they were fragmentary, perhaps more fragmentary than the term "model" might ordinarily connote: they were collections of beliefs about parts of the system, processes, or behaviors, rather than unified models of the whole design. Students would happily recruit an analogy to explain one part of the machine's operation that bore no relation to the rest of the system. This fragmentary character of mental models of complex systems may be an important aspect (see i.e., Norman 1983), allowing partial understandings to be maintained. One implication is that users' mental models of single processes or operations might be a worthwhile topic for study and practical intervention (in design or instruction).

One widely held belief about a particular process affected the students' behavior as users. Almost all respondents believed that it was not possible to type ahead during machine pauses. At the time the study was conducted this was true for some, but not all, designs in use. Consequently, in some cases transactions were presumably being needlessly slowed because of an aspect of users' mental models.

A more recent study of a similar kind is an investigation of users' models of the navigation facilities provided by Internet browsers (Cockburn and Jones 1996). Internet browsers, like Internet Explorer, maintain history lists of recently visited pages, providing direct access to these pages without needing to enter the URL or follow a hyperlink. The "back" and "forward" buttons provide a very frequently used mechanism for browsing history lists, but do users have good mental models for how they work? Cockburn and Jones (1996) showed that many do not.

The history list of visited pages can be thought of as a stack: a simple last-in-first-out data structure to which elements can be added (pushed) or taken out (popped) only from the top (consider a stack of trays in a canteen). When a new web page is visited by following a hyperlink, or by entering a URL, its address is pushed onto the top of the stack. This is true even if the page is already in the history list, so that the history list may contain more than one copy of the same page. However, when a page is visited by using the Back button (or, at least typically, by choosing from the history list), the page is not pushed onto the stack. So, what happens when the currently displayed page is not at the top of the stack (because it has been visited via the history list) and a new link is followed (or a new URL entered)? The answer is that all the pages in the history list that were above the current page are popped from the stack, and the newly visited page is pushed onto the stack in their place. For this reason the history list does *not* represent a complete record, or time-line of visited pages, and not all pages in the current browsing episode can be backed-up to. In Cockburn and Jones' study, few users appreciated this aspect of the device.

This then, has been the major thrust of work on mental models in HCI: what do people know and believe to be true about the way the systems they interact with are structured? How do their beliefs affect their behavior? In this literature a "mental model" is little more than a pointer to the relevant parts of the user's knowledge, yet this is not to deny its usefulness. One approach that it has engendered is a typology of knowledge—making groupings and distinctions about types of knowledge that are relevant in certain circumstances. It is in exactly this way that a literature on "shared mental models" as an explanatory concept in teamwork has been developed. This topic is perhaps the most rapidly growing area of mental models research in HCI and will be reviewed in the final section of this chapter.

However, as argued at length in Payne (2003), there are approaches to mental models in HCI that go beyond a concern with user knowledge and beliefs to ask more nuanced theoretical questions. The first of these is to investigate the form of mental models by inspecting the processes through which mental models might have their effects on behavior.

A powerful idea here is that mental models of machines provide a problem space that allows more elaborate encoding of remembered methods, and in which novice or expert problem solvers can search for new methods to achieve tasks.

The classic example of this approach is the work of Halasz and Moran (1983) on Reverse Polish Notation (RPN) calculators. RPN is a post-fix notation for arithmetic, so that to express 3 + 4, one would write 3 4 +. RPN does away with the need for parentheses to disambiguate composed operations. For example (1 + 2) * 3 can be expressed 1 2 + 3 * with no ambiguity. RPN calculators need a key to act as a separator between operands, which is conventionally labeled ENTER, but they do not need an = key, as the current total can be computed and displayed whenever an operator is entered.

Halasz and Moran taught one group of students how to use an RPN calculator using instructions, like a more elaborate version of the introduction above, which simply described the appropriate syntax for arithmetic expressions. A second group of subjects was instructed, using a diagram, about the stack model that underlies RPN calculation. Briefly, when a number is keyed in, it is "pushed" on top of a stack-data structure (and the top slot is displayed). The ENTER key copies the contents of the top slot down to the next slot. Any binary arithmetic operation is always performed on the contents of the top two slots and leads to the result being in the top slot, with the contents of slots 3 and below moving up the stack.

Halasz and Moran discovered that the stack-model instructions made no difference to participants' ability to solve routine arithmetic tasks: the syntactic "method-based" instructions sufficed to allow participants to transform the tasks into RPN notation. However, for more creative problems (such as calculating (6 + 4) and (6 + 3) and (6 + 2) and only keying the number 6 once) the stack group was substantially better. Verbal protocols showed that these subjects reasoned about such problems by mentally stepping through the transformations to the stack at each keystroke.

This kind of reasoning, stepping through a sequence of states in some mental model of a machine, is often called "mental simulation" in the mental models literature, and the kind of model that allows simulation is often called a "surrogate" (Young 1983; Carroll and Olson 1988). From a practical standpoint, the key property of this kind of reasoning is that it results in behavior that is richer and more flexible than the mere rote following of learned methods. The idea that the same method may be encoded more richly, so that it is more flexible and less prone to forgetting will be returned to later in the chapter when a theory of mental models of interactive artifacts is considered, and when ideas about instruction for mental models are reviewed.

A second example of mental models providing a problem space elaboration of rote methods comes in the work of Kieras and Bovair (1984). This research was similar to that of Halasz and Moran (1983) in that it compared the learning performance of two groups: (1) one instructed with rote procedures, (2) the other additionally with a diagrammatic model of the device on which the procedures were enacted. In this case, the device was a simple control panel, in which each rote procedure specified a sequence of button-pushes and knob-positions leading to a sequence of light-illuminations. The model was a circuit diagram showing the connections between power-source switches and display-lights.

Kieras and Bovair (1984) found that the participants instructed with the model learned the procedures faster, retained the procedures more accurately, executed the procedures faster, and could simplify inefficient procedures that contained redundant switch settings. They argued that this was because the model (circuit diagram) explained the contingencies in the rote-action sequences (i.e., if a switch is set to MA, so that the main accumulator circuit is selected, then the FM, fire main, button must be used).

A related theoretical idea is that mental models are a special kind of representation, sometimes called an *analog* representation: one that shares the structure of the world it represents. This was taken as the definitional property of mental models by the modern originator of the term, the British psychologist Kenneth Craik (1943). It is this intuition that encourages the use of terms like "mental simulation"—the intuition that a mental model is like a physical model, approximating the structure of what it represents, just as a model train incorporates (aspects of) the physical structure of a train.

The idea that mental models are analog in this sense is a definitional property in the work on reasoning and comprehension by Johnson-Laird (Johnson-Laird 1983, 1989; this will be further discussed in Section 3.2, concerning representations of text) and also in the theory of Holland et al. (1986) and Moray (1999). However, there are different nuances to the claim, which must be considered. And, in addition, there is a vexed question to be asked; namely, what is the explanatory or predictive force of a commitment to analog representational form? Is there any reason for HCI researchers to pay attention to theoretical questions at this level?

Certainly, this is the view of Moray (1999) who is concerned with mental models of complex dynamic systems, such as industrial plants. He proposes that models of such systems are structure-sharing *homomorphisms* rather than isomorphisms, that is, they are many to one rather than one-to-one mappings of objects, properties, and relations. (In this he follows Holland et al. 1986.)

Homomorphic models of dynamic systems may not share structure with the system at the level of static relations, but only at the level of state-changes. Thus, such models have the character of state-transition diagrams, making the empirical consequences of structure sharing somewhat unclear, because any problem space can be represented in this way.

In my view, a clearer view of the explanatory force of analog mental models can be derived by carefully considering the ideas of computational and informational equivalence first introduced by Simon (1978).

It is obviously possible to have two or more distinct representations of the same information. Call such representations "informationally equivalent" if all the information in one is inferable from the other, and vice versa. Two informationally equivalent representations may or may not additionally be "computationally equivalent," meaning that the cost structure of accessing and processing the information is equivalent in both cases, or, as Larkin and Simon (1987) put it: "information given explicitly in the one can also be drawn easily and quickly from the information given explicitly in the other, and vice versa." As Larkin and Simon point out, "easily" and "quickly" are not precise terms, and so this definition of computational equivalence is inherently somewhat vague; nevertheless it points to empirical consequences of a representation (together with the processes that operate upon it) that depend on form, and therefore go beyond mere informational content.

In Payne (2003), I propose adopting *task-relative* versions of the concepts of informational and computational equivalence. Thus, representations are informationally equivalent, *with respect to a set of tasks*, if they allow the same tasks to be performed (i.e. contain the requisite information for those tasks). The representations are, additionally, computationally equivalent with respect to the tasks they allow to be performed, if the *relative difficulty* of the tasks is the same, whichever representation is being used. (Note that according to these definitions, two representations might be computationally equivalent with regard to a subset of the tasks they support but not with regard to the total set, so that in Larkin and Simon's sense they would merely be informationally equivalent. The task-relative versions of the constructs thus allow more finely graded comparisons between representations.)

This idea can express what is behaviorally important about the idea of analog models, or structure-sharing mental representations of a state of affairs of a dynamic system. An analog representation is computationally equivalent (with respect to some tasks) to external perception and manipulation of the state of affairs it represents.

Bibby and Payne (1993, 1996) exploited this distinction between computational and informational equivalence in the domain of HCI, using a computer simulation of a device derived from that studied by Kieras and Bovair (1984). The device was a multiroute circuit, in which setting switches into one of several configurations would make a laser fire; various indicator lights showed which components of the circuit were receiving power. What concerned Bibby and Payne (1993) was the idea of computational equivalence between a mental model and a diagram of the device, rather than the device itself.

Bibby and Payne asked participants to repeatedly perform two types of tasks: a switch task, in which all but one switch was already in position to make a laser fire (the participant had to key the final switch) and a fault task, in which the pattern of indicator lights was such that one of the components must be broken (the participant had to key the name of the broken component).

Participants were instructed about the device with either a table, which showed the conditions under which each indicator light would be illuminated, or with procedures, sequences of switch positions enabling the laser to be fired. Both instructions were sufficient for both switch and fault tasks; they were informationally equivalent with respect to those tasks. However, the table made the fault task easier than the switch task, whereas the procedures made the switch task easier.

During practice, when participants consulted the instructions, this pattern of relative difficulty was confirmed by a crossover interaction in response times. Furthermore, when the instructions were removed from the participants, so that they had to rely on their mental representation of the device, the crossover interaction persevered, demonstrating that the mental representations were computationally equivalent to the external instructions.

In subsequent experiments, Bibby and Payne (1996) demonstrated that this pattern persevered even after considerable interaction with the device that might have been expected to provide an opportunity to overcome the representational constraints of the initial instruction. The crossover interaction eventually disappeared only after extended practice on the particular fault-and-switch task (80 examples of each: perhaps because of asymptotic performance having been reached). At this point, Bibby and Payne introduced two similar but new types of tasks designed so that once again, the table favored one task whereas procedures favored the other. (However, the device instructions were *not* re-presented.) At this point the crossover re-appeared, demonstrating that participants were consulting their instructionally derived mental model of the device, and that this was still in a form computationally equivalent to the original external representation of the instructions.

Practically, this research shows that the exact form of instructions may exert long-lasting effects on the strategies that are used to perform tasks, so that designers of such instructions must be sensitive not only to their informational content but also to their computational properties. In this light, they also suggest that one instructional representation of a device is very unlikely to be an optimal vehicle for supporting all user tasks: it may well be better to provide different representations of the same information, each tailored to particular tasks. In this sense, perhaps instructions should mirror and exploit the natural tendency, noted above, for users to form fragmentary mental models, with different fragments for different purposes.

In terms of theory, Bibby and Payne's findings lend support to the suggestion developed above that mental models of a device that are formed from instructions may be computationally equivalent to the external representations of the device. This idea gives a rather new reading, and one with more ready empirical consequences to the theoretically strong position that mental models are essentially analog, homomorphic representations.

3.2 MENTAL MODELS OF TEXT AND OTHER ARTIFACTS

The psychological literature on text comprehension has been transformed by the idea of a situation model, first put forward as part of a general theory of text comprehension by van Dijk and Kintsch (1983), and developed over the years by Kintsch (1998) and followers. The central idea of the general theory is that readers construct mental representations of what they read at several different levels. First, they encode the surface form of the text: the words and syntax. Second, they go beyond this to a representation of the propositional content of the text. Finally, they go beyond the propositional context of the text itself to represent what the text is about, incorporating their world knowledge to construct a situation model or mental model of the described situation.

(Under this view, it is the content that distinguishes a situation model from a text base, rather than a representational format. However, some researchers, notably Johnson-Laird (1983), and followers have pursued the idea of mental models derived from text as analog representations of the described situation. Thus, in text comprehension, there is a version of the issue discussed in part one.)

It is instructive to consider some of the evidence for situation models, and what important issues in text comprehension the theory of situation models allows us to address.

A classic early study was conducted by Bransford, Barclay, and Franks (1972). They asked participants to read simple sentences such as,

> Three turtles rested beside/on a floating log, and a fish swam beneath them.

(The slash indicates that some subjects read the sentence with the word "beside." and others read the same sentence with the word "on.")

In a later recognition test, interest centered on how likely readers were to falsely accept minor rewordings of the original sentences. In the above case, the foil sentence was

> Three turtles rested beside/on a floating log, and a fish swam beneath it.

The key finding was that people who had read the "on" versions of the sentences were much more likely to accept the changed version of the sentence, despite the fact that at the level of the sentences the difference between original and foil sentences in the two conditions is identical, limited in each case to the last word of the sentence. The reason for false recognition in one case is because, in this case, but not when "on" is replaced by "beside," the original and foil sentences describe the same situation.

A related series of experiments was reported by Fletcher and Chrysler (1990). In a series of carefully controlled experiments, they varied the overlap between sentences in a recognition test and sentences from 10 different texts read by the participants. Each text described a state of affairs (i.e., the relative cost of antiques) consistent with a linear ordering among a set of five objects. They found that participants were influenced by overlap between sentences at study and test corresponding to the three levels of discourse representation proposed by van Dijk and Kintsch (1983): surface form, text base, and situation model. Recognition performance was best when distracter items were inconsistent with all three levels of representation. Recognition was above chance when distracters violated merely the surface form of the original sentences (i.e. substituting rug for carpet). It improved further when propositional information from the text base, but not the linear ordering of the situation, was violated. Recognition was best of all when the distracters were inconsistent with the situation described by the text. This suggests that some aspects of the structure of the situation (in this case a set of linear orderings) were retained.

Next, consider work by Radvansky and Zacks (Radvansky and Zacks 1991; Radvansky, Spieler, and Zacks 1993). In these experiments, participants read sentences such as, "The cola machine is in the hotel," each of which specified the location of an object. In one condition sentences shared a common object (i.e. cola machine) but different locations. In a second condition, different objects share a common location (i.e. the city hall). Later in the experiment participants were given a speeded-recognition test. Radvansky and Zacks found a significant fan effect for the common object condition; times to verify sentences increased as the number of different locations rose. For the common location sentences no significant fan effect emerged. This was interpreted as evidence that participants formed mental models around the common location (a representation of such a location containing all the specified objects) and retrieval from long-term memory (LTM) was organized around these mental models. It is impossible, or much harder, to form such a representation of the same object in multiple locations.

What all these studies, and many like them, reveal is that when understanding text, readers spontaneously construct a mental representation that goes beyond the text itself and what it means, and use inferences to construct a richer model of what the text is about—a situation model.

Beyond these refined and clever, but undeniably rather narrow experimental contexts, the construct of situation models has been put to work to illuminate some practical issues concerning text comprehension, and exactly this issue will be returned to later, where we will see how it can inform attempts to understand instructional strategies for engendering useful mental models.

There are two principal ways in which the literature on text comprehension is relevant to HCI. First, it provides support for the idea that a mental model is a representation of what a representational artifact represents. The layered model of text comprehension previously outlined can be generalized to the claim that the user of any representational artifact must construct a representation of the artifact itself, and of what the artifact represents, and of the mapping between the two (how the artifact represents). This is the basis of the Yoked State Space (YSS) hypothesis (Payne, Squibb, and Howes 1990).

If a reader's goal is just to understand a text, as it was in the experiments just reviewed, then the text-representation can be discarded once a model has been constructed. However, there are many tasks of text *use,* in which it is necessary to maintain a representation of the text, alongside a mental model of the meaning of the text. Consider, for example, the tasks of writing and editing, or of searching for particular content in a text. In such tasks, it is necessary to keep in mind the relation between the surface form of the text—wording, spatial layout, and so on—and its meaning. Text is a representational artifact, and to *use* it in this sense one needs a mental representation of the structure of the text, and of the "situation" described by the text and of the mapping between the two.

According to the YSS hypothesis (Payne, Squibb, and Howes 1990), this requirement is general to all representational artifacts, including computer systems. To use such artifacts requires some representation of the domain of application of the artifact—the concepts the artifact allows you to represent and process. The user's goals are states in this domain, which is therefore called the goal space. However, states in the goal space cannot be manipulated directly. Instead, the user interacts with the artifact, and therefore needs knowledge of the artifact, and of the operations that allow states of the artifact to be transformed. Call this problem space the device space. In order to solve problems in the goal space by searching in the device space, the user must know how the device space represents the goal space. In this sense the two spaces need to be yoked. The minimal device space for a certain set of tasks must be capable of representing all the states in the corresponding goal space. More elaborate device spaces may incorporate device states that do not directly represent goal states, but which allow more efficient performance of tasks, just as the stack model of an RPN calculator allows an elaboration of methods for simple arithmetic.

The work of Halasz and Moran (1983) can readily be assimilated into the YSS framework. The no-model condition was provided with enough information to translate algebraic expressions into their Reverse Polish equivalent. However, in this understanding of RP expressions, the ENTER key was given merely an operational account, serving simply as a separator of operands, and did not transform the device state. The stack model, however, provides a figurative account of the ENTER key.

This discussion illustrates a practical lesson for the design of interfaces and instructions. In the case of the copy buffer and the calculator stack, the standard interface does not allow the appropriate device space readily to be induced, so that conceptual instructions must fill the gap. The obvious alternative, which has been developed to some extent in both cases, is to redesign the user interface so as to make the appropriate device space visible. These examples suggest a simple heuristic for the provision of conceptual instructions that may help overcome the considerable controversy over whether or not such instructions (as opposed to simple procedural instructions) are useful. According to this heuristic, conceptual instructions will be useful if they support construction of a YSS that the user would otherwise have difficulty inducing (Payne, Howes, and Hill 1992).

A more direct way in which text comprehension research is relevant to HCI is that so much HCI is reading text. Beyond the standard issues, the widespread availability of electronic texts raises some new concerns that have not yet seen much work, yet are perhaps the most directly relevant to HCI design. Two issues stand out: (1) the usability of documents that incorporate multiple media alongside text, and (2) the exploitation by readers of multiple texts on the same topic.

How are multimedia "texts" that incorporate graphics comprehended? There is only a small literature on this within the mainstream field of text comprehension, but this literature exploits the idea of a mental model.

Glenberg and Langston (1992) argued that the widespread idea that diagrams can assist the comprehension of technical text had, at the time, been little tested or understood and that mental models were an important explanatory construct. In their analysis, diagrams are useful in concert with texts precisely because they assist the construction of mental models. This idea has been pursued in a very active program of work on multimedia instruction by Mayer and colleagues, which will be reviewed in the next section.

What about when the multiple sources of information are not presented as part of a single text, but rather independently, covering overlapping ground, so that the reader has to perform all the integration and mapping? This is the issue of multiple texts, and it has become commonplace in the age of the Internet. It is now rarely the case that a student struggles to find relevant source documents on a topic. Instead, students are typically faced with an overabundance of relevant materials and must somehow allocate their time across them, and integrate the knowledge they derive from different sources.

Perfetti (1997) has suggested that learning from multiple texts is one of the most important new challenges for text researchers. Research has shown, for example, that integrating information across multiple texts is a skill that does not come readily but can be acquired and taught (Stahl et al. 1996; Rouet et al. 1997).

The YSS theory raises important issues here. As previously noted, everyday reading of text can be seen as engendering a progression of mental representations moving from the surface form through the propositional content to a situation model. When reading, earlier representations can be discarded as later ones are formed, but for other tasks of text use, the reader needs to maintain a representation of the form of the multitext, and map this form onto the content. Payne and Reader (in press) refer to such a representation as a structure map.

The usefulness of a structure map becomes even more apparent when multiple texts are considered. In this case, structure maps could play a role in encoding *source* information, which might be important not only for locating information, but also for integrating diverse and potentially contradictory information and for making judgments of trust or confidence in the information. Source information might additionally encode temporal properties of information sources, and thus be useful for memory updating—revising knowledge in the light of new information, making distinctions between current and superseded propositions.

The widespread availability of the web not only means that multiple texts are more widely encountered, but also encourages a situation where multiple texts are read in an interleaved fashion, in a single sitting, or at least temporally close, raising the importance of the above challenges, and meaning that recency in autobiographical memory is unlikely to accomplish source identification, so further stressing the importance of a structure map.

Payne and Reader (in press) studied readers' ability to search for specific ideas in multiple texts that they had just read. They found evidence that readers spontaneously constructed structure maps, as just described, in that they showed some memory of which documents contained which ideas, even when they did not expect to need such knowledge when reading the texts.

3.3 INSTRUCTIONS FOR MENTAL MODELS

3.3.1 Multimedia Instruction

If mental models are important for operating devices, how should they best be taught? We have seen that models are constructed automatically by readers of text, but can modern computational media, such as animations, be used to improve the acquisition of mental models from instructional texts, just as Glenberg and Langston (1992) suggested in the case of simple diagrams? Just such a question has been addressed in a long-standing program of work by Richard Mayer and colleagues, which will be reviewed in this section.

Mayer and Moreno (2002) present a cognitive theory of multimedia learning, which builds on three main ideas:

1. From dual coding theory the authors suppose that humans have separate visual and verbal information processing systems (Clark and Paivio 1991; Paivio 1986)
2. From cognitive load theory the authors assume that the processing capacity of both the visual and the verbal memory system is strictly limited (Baddeley 1992; Chandler and Sweller 1991) and that cognitive load during instruction can interfere with learning
3. From constructivist learning theory the authors take the idea that meaningful learning requires learners actively to select relevan``t information, to structure it into coherent representations, and make connections with other relevant knowledge (Mayer 1996, 1999a)

This latter process, of building coherent representations that connect information from different modalities with pre-existing knowledge, bears clear relation to Johnson-Laird's construct of mental models, and indeed Mayer and colleagues use the term in this context. In the case of the physical systems that many of their studies have addressed, mental models may take the form of cause-effect chains. According to Mayer and Moreno (2002) a key design principle for instructional materials is that they should maximize the opportunity for these model-construction processes to be completed.

Mayer and colleagues have conducted a large number of experiments comparing learning from multimedia source materials with learning from components of these materials (words, pictures, etc.) successively or in other kinds of combination. Based on this research, Mayer (1999b) and Mayer and Moreno (2002) have identified some principles of instructional design that foster multimedia learning.

The *multiple presentation principle* states that explanations in words and pictures will be more effective than explanations that use only words (Mayer and Moreno 2002, p. 107). When words only are presented, learners may find it difficult to construct an appropriate mental image, and this difficulty may block effective learning. Mayer and Anderson (1991; Experiment 2b) compared four treatment groups: (1) words with pictures, (2) words only, (3) pictures only, and (4) control, on tests of creative problem solving involving reasoning how a bicycle pump works. Results demonstrated that participants in the words with pictures group generated a greater number of creative problem solutions than did participants in the other groups. Interestingly, animation without narration was equivalent to no instruction at all. Other studies have offered support for the general idea that learners will acquire richer knowledge from narration and animation than from narration alone (Mayer and Anderson 1991, Experiment 2a; Mayer and Anderson 1992, Experiments 1 and 2).

The *contiguity principle* is the claim that simultaneous as opposed to successive presentation of visual and verbal materials is preferred (Mayer and Moreno 2002), because this will enable learners to build referential connections more readily (Mayer and Sims 1994). Mayer and Anderson (1991, Experiments 1 and 2) studied a computer-based animation of how a bicycle pump works. They compared a version that presented words with pictures against the same content presenting words before pictures, and tested acquisition with tests of creative problem solving. Those in the words-with-pictures group generated about 50% more solutions to the test problems than did subjects in the words-before-pictures group.

The *individual differences principle* predicts that factors such as prior knowledge or spatial ability will influence transfer of learning from multimedia materials, moderating the effects of other principles (Mayer 1999c). With regard to domain specific knowledge, Mayer proposed that experienced learners may suffer little decrease in problem solving transfer when receiving narration and animation successively because their background knowledge will allow a mental model to be constructed from the words alone, then linked to the visual information. Low-experience learners, on the other hand, will have no means to over-ride the effects underlying the contiguity principle, and their problem solving transfer will suffer (Mayer and Sims 1994). In support of this suggestion, experimental work by Mayer and Gallini (1990) demonstrated across three studies that the synchronization of words and pictures served to improve transfer for low- but not high-experience learners.

The *chunking principle* refers to a situation in which visual and verbal information must be presented successively, or alternately (against the contiguity principle). It states that learners will demonstrate better learning when such alternation takes place in short rather than long segments. The reasoning is straightforward, given the assumptions of the framework: working memory may become overloaded by having to hold large chunks before connections can be formed (Mayer 1999b). An experiment by Mayer and Moreno (1998) investigated this chunking principle using explanations of how lightning storms develop. The ability to solve novel, transfer problems about lightning exhibited by a 'large chunk' group (who received all the visual information before or after all the verbal information) was compared with that of a 'small chunk' group (alternating presentations of a short portion of visual followed by a short portion of narration). The gain in performance of the small chunk group over the large chunk group was circa 100% (Mayer and Moreno 1998).

The debt of Mayer's work to Sweller's program of research on Cognitive Load Theory is obvious. Mayer's design principles reflect the premise that students will learn more deeply when their visual and/or verbal memories are not overloaded. Students are better able to make sense of information when they receive both verbal and visual representations rather than only verbal; when they can hold relevant visual and verbal representations in working memory at the same time; when they have domain specific knowledge and/or high spatial ability; and when they receive small bits of information at a time from each mode of presentation.

Despite incredibly positive research results, at this stage Mayer's work should be viewed with a little caution. Almost all of the experiments utilize very short instructional presentations, with some of the animations lasting only 30 seconds. Subjects are then required to answer problem-solving questions that seem ambiguous, requiring students to be fairly creative in order to generate solutions. Mayer's work also typically neglects to include any tests of long-term retention. It may conceivably be falling into the instructional trap of maximizing performance during learning at the expense of longer-term performance. This issue is the focus of the next section.

3.3.2 Theory of Learning by Not Doing

Mayer's theory of multimedia instruction adheres to the common assumption that the optimal design of instructional material involves minimizing the cognitive burden on the learner due to the limits of the working memory.

Yet minimizing the mental effort of learners is not necessarily or always a good instructional strategy. According to Schmidt and Bjork (1992), instructional conditions that achieve the training goals of generalizability and long-term retention are not necessarily those that maximize performance during the acquisition phase.

They argue that the goal of instruction and training in real-world settings should first be to support a level of performance in the long term, and second to support the capability to transfer that training to novel-tasks environments. Methodologically, in order to measure a genuine *learning effect,* some form of long-term assessment of retention must take place; skill acquisition is not a reliable indicator of learning.

Schmidt and Bjork (1992) discussed three situations in which introducing difficulties for the learner can enhance long-term learning. First, studies that vary the scheduling of tasks during practice were reported. Random practice is more difficult than blocked schedules of practice, as a given task is never practiced on the successive trial. Using a complex motor task involving picking up a tennis ball and using it to knock over a particular set of barriers, Shea and Morgan (1979) reported a clear advantage for subjects who practiced under blocked conditions (subsets of barriers to knock), in terms of performance during practice. However, the amount of learning as demonstrated by the retention phase favored the random condition. Similar results have been reported by Baddeley and Longman (1978), Lee and Magill (1983), and (with verbal tasks) Landauer and Bjork (1978).

Schmidt and Bjork offer an explanation for this paradigm, in which retrieval practice may play a key role. They suggest that there may be a benefit, in terms of long-term retention, for activities that actually cause forgetting of the information to be recalled, forcing the learner to practice retrieving this information (Bjork and Allen 1970).

Experiments that vary the feedback the learner receives have demonstrated a similar phenomenon. A study by Schmidt et al. (1989) demonstrated that delaying the feedback that subjects received during motor tasks interfered with performance. However, on a delayed-retention test, those who had received the feedback least often demonstrated the most effective performance. This seems to contradict the established opinion that feedback is vital for effective learning. Schmidt and Bjork (1992) suggested that frequent feedback may actually serve to block information-processing activities that are important during the skill-acquisition phase.

A final area reviewed by Schmidt and Bjork concerns the introduction of variability during practice, such as when practicing tossing a beanbag at a target at a particular distance. Practicing at variable distances is more effective than practicing at a fixed distance (Kerr and Booth 1978).

Does the Schmidt and Bjork approach extend to HCI tasks, and in particular to instruction for mental models?

One impressive example of an instructional effect in the Schimdt and Bjork (1992) paradigm is informed by the idea of mental models or situation models derived from text, as discussed in Section 3.2 of this chapter. Informed by the distinction between a text base and a situation model, work by McNamara et al. (1996) has shown how expository text can be designed to introduce difficulties for readers in exactly the productive manner advocated by the Schmidt and Bjork conception of training. These authors created two versions of target texts, one more coherent than the other (one experiment used a text about traits of mammals, a second used a text about heart disease). Coherence cues were provided by linking clauses with appropriate connectives and by inserting topic headings. The level of readers' background knowledge on the topic of the text was also assessed with a pretest. After reading a text, participants were given tests of the text base (free recall of the text propositions and specific factual questions about the contents of the text) and tests of the situation model (problem-solving-based questions, questions requiring inferences from the text, and a concept-sorting task).

McNamara et al. (1996) reported that for measures that tested the text base, the high coherence texts produced better performance. However, for situation-model measures, test performance for high-knowledge readers was better when they read the low-coherence text. McNamara et al. argued that limiting the coherence of a text forced readers to engage in compensatory processing to infer unstated relations in the text. This compensatory processing supported a deeper understanding of the text, in that the information in the text became more integrated with background knowledge. Thus, for high-knowledge readers, the texts that were more difficult to read improved the situation model by encouraging more transfer-appropriate processing. Low-knowledge readers were, presumably, unable to achieve the compensatory inferences, and therefore did better with more coherent texts. Because the text base does not incorporate background knowledge, it was not enhanced by any compensatory processing. (This finding is related to the work of Mayer and Sims [1994] reviewed above.)

One very successful practical approach to the design of instructions for interactive devices which is well known in the HCI community, is perhaps quite strongly related to this more theoretically oriented work. The concept of a "minimal manual" was outlined by Carroll (1990). It sought to minimize the extent to which instructional materials obstruct learning. Crucially, a well-designed Minimal Manual does not necessarily optimize the speed at which users can perform procedures as they read. Carroll's manuals avoided explicit descriptions that encouraged rapid but mindless rote performance. Instead, the emphasis was on active learning whereby learners were encouraged to generate their own solutions to meaningful tasks. This process was facilitated in part by reducing the amount of text provided and including information about error recovery.

O'Hara and Payne (1998, 1999) argued that learning from a problem-solving experience might be enhanced to the extent that problem solvers planned their moves through the problem space. Many puzzles with an interactive user interface, and indeed many user interfaces to commercial systems, encourage a one-step-at-a-time approach to problem solving, in which a move is chosen from the currently available set. This may be quick and relatively effortless, yet lead to little learning and inefficient solutions. For example, in an HCI task, participants had to copy names and addresses from a database to a set of letters. Each item category from the database had to be copied to several letters, so that the most obvious and perhaps least effortful strategy of preparing letters one at a time was inefficient in terms of database access. O'Hara and Payne's manipulation was to increase the cost of making each move (in the copying experiment by adding a system lock-out time). This resulted in more planning, more think-time per move, meaning slower solutions in the first instance, but more efficient behavior in the long term, and the discovery of strategies that required fewer database accesses and fewer user inputs.

Recent work by Duggan and Payne (2001) combined several of the insights in the work just reviewed to explore acquisition of interactive procedures during instruction following. Good procedural instructions for interactive devices must satisfy two criteria. First, they must support performance. Like all procedural instructions they should effectively communicate the procedure they describe, so as to allow users who don't know the procedure to enact it successfully and efficiently. Second, they must support learning. In common with instructions for all procedures that will be used repeatedly, they should facilitate subsequent memory for the procedure, so that it might later be performed without consulting the instructions.

How might procedural instructions be designed so as to follow the Schmidt and Bjork paradigm and provide transfer-appropriate practice opportunities for the learner? Of course, not all manipulations that introduce difficulties during learning are beneficial for the learner. Simply making the instructions unclear is unlikely to be effective. However, much this idea may have informed the design of some commercial user manuals. The criterion that quality instructions must communicate the procedure that they describe cannot be ignored.

The work of Diehl and Mills (1995) further illustrated the relevance of the theory of text comprehension to the design of instruction for interactive procedures. They argued that in the case of procedural instructions the distinction between situation model and text base maps directly onto a distinction between memory for the procedure (as tested by later task performance) and memory for the instructions themselves.

Texts describing how to complete a task using a device (setting an alarm clock or constructing a child's toy) were provided. While reading a text, participants were required to either perform the task (read and do), or do nothing (read only). (In addition, Diehl and Mills studied some intermediate conditions, such as read and watch experimenter. These conditions produced intermediate results and are not relevant to the current argument.) The effect of these training methods was then examined by asking participants to recall the text and then complete the task.

Diehl and Mills reported that the increased exposure to the device in the read-and-do condition resulted in improved task performance times relative to the read-only condition. However, text recall was better in the read-only condition, supporting the conceptual separation of text base and situation model.

Inspired by this work, Duggan and Payne (2001) introduced a particular technique to exploit the principle of Schmidt and Bjork (1992) and the methods of McNamara and colleagues (1996). Like the manipulations of Diehl and Mills (1995), their innovation centered not on the design of the instructions per se, but rather on the way the instructions are read and used. Diehl and Mills' reported advantage for reading and doing over reading alone has no real practical implication, as it is difficult to imagine anyone advocating isolated reading as a preferred method. However, Duggan and Payne suggested that the way learners manage the interleaving of reading and doing will affect their later retention, and thus offers an important lever for improving instruction.

Many procedural instructions have a natural step-wise structure, and in these cases it is possible to execute the procedure while reading with minimal load on memory. Learners can read a single step, and then execute it before reading the next step. Such an approach is low in effort (and therefore attractive to the learner), but also low in transfer-appropriate practice and therefore, one would argue on the basis of the reviewed work, poor at encouraging retention. If learners could instead be prompted to read several procedural steps before enacting them, performance would be made more effortful, but learning might benefit. Readers would be encouraged to integrate the information across the chunk of procedural steps, and the increased memory load would provide transfer-appropriate practice.

Duggan and Payne (2001) developed this idea as follows. First, by implementing an online help system in the context of experimental tasks (programming a VCR) they forced participants into either a step-wise or a chunk-based strategy for interleaving reading and acting. These experiments demonstrated that reading by chunks did tax performance during training, but improved learning, in particular retention of the procedure. Next, they developed a more subtle, indirect manipulation of chunking. By adding a simple cost to the access of online instructions (c.f., O'Hara and Payne 1998), they encouraged readers to chunk steps so as to minimize the number of times the instructions were accessed. Just as with enforced chunking, this led to improved retention of the procedures.

3.4 SHARED MENTAL MODELS

In the last 10 years or so there has been a rapid surge of interest in the concept of shared mental models in the domain of teamwork and collaboration. The use of mental models in this literature, to date, is somewhat inexact, with little theoretical force, except to denote a concern with what the team members know, believe, and want. As the name suggests, shared mental models refers to the overlap in individuals' knowledge and beliefs.

The central thesis and motive force of the literature is that team performance will improve when team members share relevant knowledge and beliefs about their situation, task, equipment, and team. Different investigations and different authors have stressed different aspects of knowledge, and indeed proposed different partitions into knowledge domains. (And recently, as we shall see, some investigators have questioned the extent to which overlapping knowledge is a good thing. There are some situations in which explicit distribution or division of knowledge may serve the team goals better.)

At first glance, the idea that teams need to agree about or share important knowledge seems intuitively plain. Models of communication (i.e., Clark 1992) stress the construction of a common ground of assumptions about each partner's background and intentions. The idea of shared mental models develops this idea in a plausible practical direction.

A recent study by Mathieu et al. (2000) was one of the most compelling demonstrations of the basic phenomenon under investigation, as well as being centered on an HCI paradigm. For these reasons, this study will be described and used as a framework to introduce the space of theoretical and empirical choices that characterize the mainstream of the shared mental models literature.

Mathieu et al. (2000) considered team members' mental models as comprising knowledge of four separate domains: (1) technology (essentially the mental models described in part one of this chapter); (2) job or task; (3) team interaction (such as roles, communication channels and information flow), and (4) other teammates' knowledge and attitudes. Knowledge of the last three types would rarely be called a mental model outside this literature, and so straight away we can see a broader and more practical orientation than in individually oriented mental models literatures.

For the purposes of operationalization, the authors suggested that these four categories of knowledge may be treated as two: task related and team related. This binary distinction mirrors a distinction that has been made in terms of team behaviors and communications, which have been considered in terms of a task track and a teamwork track (McIntyre and Salas 1995).

Mathieu and colleagues studied dyads using a PC-based flight simulator. One member of each dyad was assigned to the joystick to control aircraft position. The other was assigned to keyboard, speed, weapon systems, and information gathering. Both members could fire weapons. The experimental procedure incorporated a training phase, including the task and basics of teamwork, and then the flying of six missions, divided into three equally difficult blocks of two, each mission lasting around 10 minutes. Performance on a mission was scored in terms of survival, route following, and shooting enemy planes. Team processes were scored by two independent raters viewing videotapes to assign scores, for example, how well the dyad communicated with each other.

Mental models were measured after each pair of missions. At each measurement point, each individual's task or team mental model was elicited by the completion of a relatedness matrix (one for task, one for team), in which the team member rated the degree to which each pair from a set of dimensions was related. For the task model there were eight dimensions, including diving versus climbing; banking or turning; and choosing airspeed. For the team model there were seven dimensions, including amount of information and roles and team spirit.

Thus, at each measurement point, participants had to assign numbers between –4 (negatively related, a high degree of one requires a low degree of the other) and +4 (positively related, a high degree of one requires a high degree of the other) to each pair of dimensions in each domain. For example, they had to rate the relatedness of diving versus climbing to choosing airspeed, and the relatedness of roles to team spirit. For each team at each time for each model-type a convergence index was calculated by computing a correlation co-efficient (QAP correlation) between the two matrices. The co-efficient could vary from –1 (complete disagreement) to +1 (completely shared mental models).

The main findings of this investigation were as follows. Contrary to hypothesis, convergence of mental models did not increase over time; rather it was stable across missions 1 to 3. This runs counter to a major and plausible assumption of the shared mental models program, which is that agreement between team members should increase with extent of communication and collaboration.

Nevertheless, convergence of both task and team models predicted the quality of team process and the quality of performance. Further, the relationship between convergence and performance was fully mediated by quality of team process.

The most natural interpretation of these findings is that team process is supported by shared mental models. In turn, good team processes lead to good performance. According to its authors, this study provided the first clear empirical support for the oft-supposed positive relationship between shared mental models and team effectiveness (Mathieu et al. 2000, p. 280).

As well as being paradigmatic in illustrating the key ideas in the shared mental models literature, this study has several aspects that highlight the range of approaches and the controversy in the field.

First, it is worth considering what particular properties of the task and teams may have contributed to the positive relation between shared mental models and team process and performance. Compared with most situations in which coordination and collaboration are of prime interest, including most situations addressed by CSCW researchers, the teams studied by Matheiu et al. were minimal (two members) and the tasks were very short term and relatively circumscribed. Beyond these obvious remarks, I would add that the division of labor in the task was very "close," and the workers' performance was extremely interdependent. Of course, interdependence is the signature of collaborative tasks; nevertheless, a situation in which one person controls airspeed and another controls altitude may make this interdependence more immediate than is the norm.

It is also possible that the relatively circumscribed nature of the task and collaboration contributed to the failure of this study to find evidence for the sharing of mental models increasing across the duration of collaboration.

As just mentioned, although the literature contains many proposals that shared mental models will positively influence process and performance, there has been much less empirical evidence. Another study of particular relevance to HCI is concerned with the workings of software development teams.

Software development is an ideal scenario for the study of team coordination for several reasons. First, much modern software development is quintessentially team based (Crowston and Kammerer 1998; Curtis, Krasner, and Iscoe 1998; Kraut and Streeter 1995), and relies heavily on the complex coordinations of team members. Secondly, this effort is often geographically dispersed, further stressing collaboration and putting an emphasis on communications technologies. Finally, software development takes place in technologically advanced settings with technologically savvy participants, so that it provides something of a test bed for collaboration and communication technologies.

One study of complex geographically distributed software teams has been reported that partially supports the findings of the Mathieu et al. (2000) study and provided complementary evidence for positive effects of shared mental models on team performance. Espinosa et al. (2002) reported a multimethod investigation of software teams in two divisions of a multinational telecommunications company. The most relevant aspect of their study was a survey of 97 engineers engaged in team projects of various sizes ranging from 2 to 7. Team coordination and shared mental models (SMM) were both measured by simple survey items, followed by posthoc correlational analysis to uncover the relation between shared mental models and team process. As in the Mathieu et al. (2000) study, shared mental models were considered in two categories: task and team. A positive relation between team SMM and coordination was discovered, but the effect of task SMM was not significant.

It is worth being clear about the positive relation and how it was computed. Team SMM was computed for each team by assessing the correlations between each team member's responses to the team SMM survey items. This index was entered as an independent variable in a multiple regression to predict average reported levels of team coordination. It is, of course, hard to infer any causal relation from such correlational analyses, and one might also wonder about the validity of purely questionnaire-based measures of some of the constructs, yet nevertheless the study is highly suggestive that SMM can have a positive influence in group-work situations far removed from pairs of students interactive with a flight simulator. Additionally, Espinosa et al. (2002) reported an interview study in which respondents confirmed their own belief that SMM contributed positively to project communications and outcomes.

Nevertheless, Espinosa et al. (2002) failed to find any relation between task SMM and team process. It seems to me that, in view of the survey methodology, this would have been the more compelling evidence in favor of the SMM construct. It seems less surprising and perhaps less interesting that there should be a correlation between participants' survey responses concerning how well they communicated on their team, and, for example, their agreement about which teammates had high knowledge about the project.

Levesque, Wilson, and Wholey (2001) reported a different study of software development teams, using ad hoc student-project groupings to study whether sharing of Team SMM increased over time. They only measured Team SMM, using Likert scale items on which participants signaled amount of agreement or disagreement with statements like, “Most of our team’s communication is about technical issues,” “Voicing disagreement on this team is risky,” or “Lines of authority on this team are clear.” Team SMM was measured by computing correlations among team members of these responses after 1, 2, and 3 months of working on a joint project.

Levesque, Wilson, and Wholey (2001) found that, contrary to their hypothesis, team SMM decreased over time. They argue that this is because projects were managed by a division of labor that required much initial collaboration but meant that later activity was more individual.

There are surely many teamwork situations in which role differentiation is critical for success, and this observation suggested that the most straightforward interpretation of shared mental models is overly simple. Indeed, even in teams that continue to meet, communicate, and collaborate, it may be that role differentiation means that task mental models should not so much be “shared” as “distributed” to allow for effective team performance. (Studies of intimate couples have explored a similar process of specialization of memory functions, under the name “transactive memory,” i.e., Wegner 1987, 1995).

When roles are differentiated, it is no longer important that task knowledge is shared, but rather that individuals’ knowledge about who knows what is accurate. Thus, one would expect team SMMs to support communication and collaboration even in teams with highly differentiated roles. This may explain the previously reviewed findings. In the Mathieu et al. study, the team members’ technical roles remained tightly interdependent, so that both task and team models had to be shared for successful performance. In the Espinosa et al. (2002) study, the technical roles may have been differentiated but the level of communication remained high, so that team SMM affected performance but task SMM did not. In the Levesque et al. study, the teams divided their labor to the extent that communication and collaboration ceased to be necessary (apart, perhaps for some final pooling of results). In this case, we would predict that neither task nor team SMMs would affect performance once the division of labor had been accomplished. No data on performance were reported, but team models became less shared over the course of the projects.

Although there has been quite a sudden flurry of interest in shared mental models, this brief review makes clear that much empirical and conceptual work remains to be done. Of particular relevance to this chapter is the question of what exactly is meant by a mental model in this context.

To date throughout the field, mental models have been considered as semantic knowledge, using traditional associative networks as a representation. Thus, mental models have typically been tapped using simple likert scales or direct questions about the relations (i.e. similarity) between constructs, analyzed with multidimensional techniques such as pathfinder (for a review of measurement techniques in this field, see Mohammed, Kilmoski, and Rentsch 2000). Because interest has focused on the extent to which knowledge and beliefs are common among team members, these approaches have been useful, allowing quantitative measures of similarity and difference. Nevertheless, compared with the literature on individual mental models, they tend to reduce participants’ understanding to mere associations, and yet the thrust of the individual work shows that this may not be appropriate, because the particular conceptualizations of the domain, the analogies drawn, the computational as well as informational relations between internal and external representations, and so on can have real effects on performance. It seems that an important path of development may be to adopt this more refined cognitive orientation and investigate the impact of shared models—as opposed to shared networks of facts and associations—on collaboration.

REFERENCES

Baddeley, A. 1992. Working memory. *Science* 255:556–9.

Baddeley, A. D., and D. J. A. Longman. 1978. The influence of length and frequency of training session on the rate of learning to type. *Ergonomics* 21:627–35.

Bibby, P. A., and S. J. Payne. 1993. Internalization and use–specificity of device knowledge. *Hum Comput Interact* 8:25–56.

Bibby, P. A., and S. J. Payne. 1996. Instruction and practice in learning about a device. *Cognitive Sci* 20:539–78.

Bjork, R. A., and T. W. Allen. 1970. The spacing effect: Consolidation or differential encoding? *J Verbal Learn Verbal Behav* 9:567–72.

Bransford, J. D., J. R. Barclay, and J. J. Franks. 1972. Sentence memory: A constructive versus interpretive approach. *Cogn Psychol* 3:193–209.

Carroll, J. M. 1990. *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill.* Cambridge, MA: MIT Press.

Carroll, J. M., and J. R. Olson. 1988. Mental models in human–computer interaction. In *Handbook of Human–Computer Interaction*, ed. M. Helander, 45–65. New York: Elsevier.

Chandler, P., and J. Sweller. 1991. Cognitive load theory and the format of instruction. *Cogn Instruct* 8:293–332.

Clark, H. H. 1992. *Arenas of Language Use*. Chicago, IL: Chicago University Press.

Clark, J. M., and A. Paivio. 1991. Dual coding theory and education. *Educ Psychol Rev* 3(3):149–70.

Cockburn, A., and S. Jones. 1996. Which way now? Analysing and easing inadequacies in WWW navigation. *Int Hum Comput Stud* 45:195–30.

Collins, A., and D. Gentner. 1987. How people construct mental models. In *Cultural Models in Language and Thought*, ed. D. Holland and N. Quinn. Cambridge, UK: Cambridge University Press.

Craik, K. J. W. 1943. *The Nature of Explanation.* Cambridge: Cambridge University Press.

Crowston, K., and E. E. Kammerer. 1998. Coordination and collective mind in software requirements development. *IBM Syst J* 37(2):227–45.

Curtis, B., H. Krasner, and N. Iscoe. 1988. A field study of the software design process for large systems. *Commun ACM* 31(11):1268–86.

Diehl, V. A., and C. B. Mills. 1995. The effects of interaction with the device described by procedural text on recall, true/false, and task performance. *Mem Cogn* 23(6):675–88.

Duggan, G. B., and S. J. Payne. 2001. Interleaving reading and acting while following procedural instructions. *J Exp Psychol Appl* 7(4):297–307.

Ericsson, K. A., R. T. Krampe, and C. Tesch-Romer. 1993. The role of deliberate practice in the acquisition of expert performance. *Psychol Rev* 100(3):363–406.

Espinosa, J. A., R. E. Kraut, S. A. Slaughter, J. F. Lerch, J. D. Herbsleb, and A. Mockus. 2002. Shared mental models, familiarity and coordination: A multi-method study of distributed software teams. In *Proceedings of the 23rd International Conference in Information Systems (ICIS)*, 425–33. Barcelona, Spain.

Fletcher, C. R., and S. T. Chrysler. 1990. Surface forms, textbases and situation models: Recognition memory for three types of textual information. *Discourse Process* 13:175–90.

Gentner, D., and A. L. Stevens. 1983. *Mental Models*. Hillsdale, NJ: Erlbaum.

Glenberg, A. M., and W. E. Langston. 1992. Comprehension of illustrated text: Pictures help to build mental models. *J Mem Lang* 31:129–51.

Halasz, F. G., and T. P. Moran. 1983. Mental models and problem-solving in using a calculator. In *Proceedings of CHI 83 Human Factors in Computing Systems.* New York: ACM.

Holland, J. H., K. J. Holyoak, R. E. Nisbett, and P. R. Thagard. 1986. *Induction*. Cambridge, MA: MIT Press.

Johnson-Laird, P. N. 1983. *Mental Models*. Cambridge, UK: Cambridge University Press.

Johnson-Laird, P. N. 1989. Mental models. In *Foundations of Cognitive Science,* ed. M. I. Posner. Cambridge, MA: MIT Press.

Kerr, R., and B. Booth. 1978. Specific and varied practice of a motor skill. *Percept Mot Skills* 46:395–401.

Kieras, D. E., and S. Bovair. 1984. The role of a mental model in learning to use a device. *Cogn Sci* 8:255–73.

Kintsch, W. 1998. *Comprehension*. Cambridge, UK: Cambridge University Press.

Kraut, R. E., and L. A. Streeter. 1995. Coordination in software development. *Commun ACM* 38(3):69–81.

Landauer, T. K., and R. A. Bjork. 1978. Optimum rehearsal patterns and name learning. In *Practical Aspects of Memory,* ed. M. M. Gnineberg, P. E. Morris, and R. N. Sykes, 625–32. London: Academic Press.

Larkin, J. H., and H. A. Simon. 1987. Why a diagram is (sometimes) worth ten thousand words. *Cogn Sci* 11:65–100.

Lee, T. D., and R. A. Magill. 1983. The locus of contextual interference in motorskill acquisition. *J Exp Psychol: Learn Mem Cogn* 9:730–46.

Levesque, L. L., J. M. Wilson, and D. R. Wholey. 2001. Cognitive divergence and shared mental models in software development project teams. *J Organ Behav* 22:135–44.

Mathieu, J. E., T. S. Heffner, G. F. Goodwin, E. Salas, and J. A. Cannon-Bowers. 2000. The influence of shared mental models team process and performance. *J Appl Psychol* 85:273–83.

Mayer, R. E. 1996. Learning strategies for making sense out of expository text: The SOI model for guiding three cognitive processes in knowledge construction. *Educ Psychol Rev* 8:357–71.

Mayer, R. E. 1999a. Research-based principles for the design of instructional messages: The case of multimedia explanations. *Doc Des* 1:7–20.

Mayer, R. E. 1999b. Multimedia aids to problem solving transfer. *Int J Educ Res* 31:611–23.

Mayer, R. E. 1999c. Multimedia aids to problem-solving transfer. *Int J Educ Res* 31:661–24.

Mayer, R. E., and R. B. Anderson. 1991. Animations need narrations: An experimental test of a dual-coding hypothesis. *J Educ Psychol* 83: 484–90.

Mayer, R. E., and R. B. Anderson. 1992. The instructive animation: Helping students build connections between words and pictures in multimedia learning. *J Educ Psychol* 84:444–52.

Mayer, R. E., and J. K. Gallini. 1990. When is an illustration worth ten thousand words? *J Educ Psychol* 82:715–26.

Mayer, R. E., and R. Moreno. 1998. A split-attention affect in multimedia learning: Evidence for dual processing systems in working memory. *J Educ Psychol* 90:312–20.

Mayer, R. E., and R. Moreno. 2002. Aids to computer-based multimedia learning. *Learn Instruct* 12:107–19.

Mayer, R. E., and V. K. Sims. 1994. For whom is a picture worth a thousand words? Extensions of a dual-coding theory of multimedia learning. *J Educ Psychol* 86:389–401.

Mayhew, D. J. 1992. *Principles and Guidelines in Software User Interface Design*. Englewood Cliffs, NJ: Prentice Hall.

McCloskey, M. 1983. Naïve theories of motion. In *Mental Models,* ed. D. Gentners, and A. L. Stevens, 299–323. Hillsdale, NJ: Erlbaum.

McIntyre, R. M., and E. Salas. 1995. Measuring and managing for team performance: Emerging principles from complex environments. In *Team Effectiveness and Decision Making in Organizations,* ed. R. A. Guzzo and E. Salas, 9–45. San Francisco, CA: Jossey-Bass.

McNamara, D. S., E. Kintsch, N. B. Songer, and W. Kintsch. 1996. Are good texts always better? Text coherence, background knowledge, and levels of understanding in learning from text. *Cogn Instruct* 14:1–43.

Moheammet, S., and B. C. Dunville. 2001. Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries. *J Organ Behav* 22(2):89.

Mohammed, S., R. Klimoski, and J. R. Rentsch. 2000. The measurement of team mental models: We have no shared schema. *Organ Res Methods* 3(2):123–65.

Moray, N. 1999. Mental models in theory and practice. In *Attention and performance XVII*, ed. D. Gopher and A. Koriat, 223–58. Cambridge, MA: MIT Press.

Newell, A., and H. A. Simon. 1972. *Human Problem Solving*. Englewood Cliffs, NJ: Prentice Hall.

Norman, D. A. 1983. Some observations on mental models. In *Mental Models*, ed. D. Gentner and A. L. Stevens, 7–14. Hillsdale, NJ: Erlbaum.

O'Hara, K. P., and S. J. Payne. 1998. The effects of operator implementation cost on planfulness of problem solving and learning. *Cogn Psychol* 35:34–70.

O'Hara, K. P., and S. J. Payne. 1999. Planning and user interface: The effects of lockout time and error recovery cost. *Int Hum Comput Stud* 50:41–59.

Paivio, A. 1986. *Mental Representations.* New York: Oxford University Press.

Payne, S. J. 1991. A descriptive study of mental models. *Behav Inf Technol* 10:3–21.

Payne, S. J. 2003. Users' mental models of devices: The very ideas. In *HCI Models, Theories and Frameworks*: *Towards a Multidisciplinary Science*, ed. J. M. Carroll, 135–56. San Francisco, CA: Morgan Kaufmann.

Payne, S. J., A. Howes, and E. Hill. 1992. Conceptual instructions derived from an analysis of device models. *Int J Hum Comput Interact* 4:35–58.

Payne, S. J., and W. R. Reader. 2006. Constructing structure maps of multiple on-line texts. *Int J Hum Comput Stud* 64(5):461–74.

Payne, S. J., H. R. Squibb, and A. Howes. 1990. The nature of device models: The yoked state space hypothesis and some experiments with text editors. *Hum Comput Interact* 5:415–44.

Perfetti, C. A. 1997. Sentences, individual differences, and multiple texts. Three issues in text comprehension. *Discourse Process* 23:337–55.

Radvansky, G. A., D. H. Spieler, and R. T. Zacks. 1993. Mental model organization. *J Exp Psychol Learn Mem Cogn* 19:95–114.

Radvansky, G. A., and R. T. Zacks. 1991. Mental models and fact retrieval. *J Exp Psychol Learn Mem Cogn* 17:940–53.

Rouet, J. F., M. Favart, M. A. Britt, and C. A. Perfetti. 1997. Studying and using multiple documents in history: Effects of discipline expertise. *Cogn Instruct* 75(1):85–106.

Schmidt, R. A., and R. A. Bjork. 1992. New conceptualizations of practice: Common principles in three paradigms suggest new concepts for training. *Psychol Sci* 3:207–17.

Schmidt, R. A., D. E. Young, S. Swinnen, and D. C. Shapiro. 1989. Summary knowledge of results for skill acquisition: Support for the guidance hypothesis. *J Exp Psychol Learn Mem Cogn* 15:352–9.

Shea, J. B., and R. L. 1979. Contextual interference effects on the acquisition, retention, and transfer of a motor skill. *J Exp Psychol Hum Learn Mem* 5:179–87.

Simon, H. A. 1955. A behavioral model of rational choice. *Q J Econ* 69:99–118.

Simon, H. A. 1978. On the forms of mental representation. In *Minnesota Studies in the Philosophy of Science*, ed. C. W. Savage, 3–18, Vol. 9. Minneapolis, MN: University of Minnesota Press.

Simon, H. A. 1992. What is an "explanation" of behavior? *Psychol Sci* 3:150–61.

Stahl, S. A., C. R. Hind, B. K. Britton, M. M. McNish, and D. Bosquet. 1996. What happens when students read multiple source documents in history. *Read Res* 31(4):430–56.

Tognazzini, B. 1992. *Tog on Interface*. Reading, MA: Addison-Wesley.

van Dijk, T. A., and W. Kintsch. 1983. *Strategies of Discourse Comprehension*. New York: Academic Press.

Wegner, D. M. 1987. Transactive memory: A contemporary analysis of the group mind. In *Theories of Group Behaviour*, ed. I. B. Mullen and G. R. Goethals, 185–208. New York: Springer-Verlag.

Wegner, D. M. 1995. A computer network model of human transactive memory. *Soc Cogn* 13:319–39.

Young, R. M. 1983. Surrogates and mappings. Two kinds of conceptual models for interactive devices. In *Mental Models*, ed. D. Gentner and A. L. Stevens, 35–52. Hillsdale, NJ: Erlbaum.

4 Task Loading and Stress in Human–Computer Interaction

Theoretical Frameworks and Mitigation Strategies

James L. Szalma, Gabriella M. Hancock, and Peter A. Hancock

CONTENTS

Individuals whose professional lives revolve around human–computer interaction (HCI) might well ask themselves why they should even glance at a chapter on stress. It is evident that many computer systems have to support people operating in stressful circumstances and, of course, there are important design issues concerning how to present information in these very demanding circumstances. However, one can legitimately question whether such issues are of any interest to those operating in mainstream HCI. Indeed, if these were the only issues we would most probably agree and recommend the reader to pass on quickly to something of much more evident relevance. However, we hope to persuade the reader that the various aspects of stress research and its application to HCI are not limited to such concerns alone. Indeed, we hope to convince the reader that stress, in its critical form of task loading, is central to all HCIs. To achieve this goal, we first present a perspective that puts stress front and center in the HCI realm. Traditionally, stress has been considered to result from exposure to some adverse environmental circumstances such as excessive heat, cold, noise, or vibration (Hancock, Ross, and Szalma 2007; Conway, Szalma, and Hancock 2007). Its effects manifest themselves primarily in relation with physiological responses most perturbed by the stress at hand. However, Hancock and Warm (1989) observed that stress effects are virtually all mediated through the brain; but for the cortex such effects are almost always of secondary concern since the brain is primarily involved with the goals of ongoing behavior or more simply with dealing with the current task at hand (see Hancock 2010). Therefore, we want to change the orientation of concern here so that stress is not just a result of external interference but rather the primary source of stress comes from the *ongoing task itself*. As we now view the task itself as the primary driving influence, then stress concerns are manifestly and evidently *central to all HCI issues*.

It is one of the clearest paradoxes of modern work that computer-based systems designed to reduce task complexity and cognitive workload actually often impose even greater demands and stresses on the very individuals they are supposed to be helping. Think of how many times in your own work that the computer has appeared to be a barrier to task completion rather than a helpful tool. How individuals cope with such stress has both immediate and prolonged effects on their performance and well-being. Although operational environments and their associated tasks vary considerably (e.g., air traffic control, baggage screening, hospital patient monitoring, power plant operations, command and control, banking/finance, and general office work), there are certain mechanisms that are common to all stress appraisals and thus to all task demands. Consequently, there are design and HCI principles to address the stress of task demand that can be generalized across many, if not all, domains (Hancock and Szalma 2003a,b). In this chapter we explore these principles to further understand and even exploit stress effects in the HCI domain.

The structure of this chapter flows from these fundamental observations: First, we provide the reader with a brief overview of stress theory and its historical development to set our observations in context. Then we articulate areas for future research, which is needed to more completely understand how stress and workload impact HCI and how their positive effects can be exploited while mitigating their negative effects. We conclude by providing an overview of these principles and some directions for future effort.

4.1 TRADITIONAL APPROACHES TO STRESS RESEARCH

As we have seen, stress has traditionally been conceived of as either an external, aversive stimulus (constituted of physical, cognitive, or social stimulation patterns) imposed on an individual or that person's individual response to such perturbations. Each of these theoretical perspectives has limited explanatory power. Considering stress as an external stimulation is useful for categorizing effects of physical environments (e.g., heat, noise, vibration), but such an approach cannot explain why the same stimulus pattern produces vastly different effects on different individuals. Physiological interpretations (e.g., Selye 1976) have utilized arousal explanations of stress. However, more recent demonstrations that different sources of stress are associated with different patterns of cognitive effects make it clear that adaptation or so-called arousal theories of stress cannot, by themselves, completely address the issue (Hockey 1984; Hockey and Hamilton 1983; Hockey, Gaillard, and Coles 1986).

Thus, to understand stress effects we now have to embrace an even wider, multidimensional perspective (e.g., Matthews

2001). In this chapter, we choose to emphasize a view of stress as primarily an outcome of the appraisal of environmental demands that either tax or exceed an individual's resources to cope with that demand. These person–environment *transactions* (Lazarus and Folkman 1984) occur at multiple levels within an organism (Matthews 2001; van Reekum and Scherer 1997). Further, these processes represent efforts by the organism to adapt to imposed demands via regulation of its own internal state while seeking to change the external environment (e.g., obtaining shelter). In Section 4.2, we describe the theoretical frameworks that guide our observations on stress in the context of HCI. These perspectives emerge from the work of Lazarus (1999; and see also Lazarus and Folkmanm 1984), Hancock and Warm (1989), and Hockey (1997).

4.2 THEORETICAL FRAMEWORKS

Herein is a brief introduction to key theories regarding stress and its effects on performance.

4.2.1 Appraisal Theory

Among the spectrum of cognitive theories of stress and emotion, perhaps the best known is the "cognitive–motivational–relational" theory proposed by Richard Lazarus and his colleagues (see Lazarus 1991, 1999; Lazarus and Folkman 1984). This theory is cognitive in that stress and emotion each depends on an individual's cognitive appraisals of internal and external events. These appraisals in their turn depend in part on the person's knowledge and experience (cf. Bless 2001). The theory is motivational in that emotions in general, including stress responses, are reactions to one's perceived state of progress toward or away from one's goals (see Carver and Scheier 1998). The relational aspect emphasizes the importance of the transaction between individuals and their environment. Together these three components shape the emotional and stress state of an individual. The outcomes of these processes are patterns of appraisal that Lazarus (1991) refers to as "core relational themes." For instance, the core relational theme for anxiety is uncertainty and existential threat, whereas that for happiness is evident progress toward goal achievement. Thus, when individuals appraise events relative to their desired outcomes (goals), negative, "goal-incongruent" emotions and stress can be produced if such events are appraised as hindering progress. Conversely, promotion of well-being and pleasure occurs when events are appraised as facilitating progress toward a goal (i.e., goal-congruent emotions). Promotion of pleasure and happiness (see Hancock, Pepe, and Murphy 2005; Ryan and Deci 2001), therefore, requires the design of environments and tasks themselves that afford goal-congruent emotions. The understanding of interface characteristics in HCI that facilitate positive appraisals and reduce negative appraisals is thus a crucial issue and an obvious avenue in which HCI and stress research can fruitfully interact.

A major limitation of all appraisal theories, however, is neglecting to understand how task parameters influence resulting coping response. Although the appraisal mechanism itself may be similar across individuals and contexts (e.g., see Scherer 1999), the specific content (e.g., which events are appraised as a threat to well-being) obviously varies across individuals and contexts. One would expect that the criteria for appraisal (e.g., personal relevance, self-efficacy for coping) are similar across individuals for specific task parameters as for any other stimulus or event. However, individual differences occur in the specific content of an appraisal (e.g., one person's threat is another's challenge) and therefore in the resultant response. An understanding of stress effects in HCI thus requires understanding the task and person factors and treating the transaction between the human being and the system as the primary unit of analysis (see Lazarus and Folkman 1984). This entails knowing how different individuals appraise specific task parameters and how changes in knowledge structures might ameliorate negative stress effects and promote adaptive affective states in human–technology interaction. A visual representation of this emergent unit of analysis that comes from the interaction of a person and the environment, including the task, is shown in Figure 4.1 (Hancock 1997).

4.2.2 Adaptation under Stress

A theoretical framework developed specifically for stress as it relates to performance is the maximal adaptability model presented by Hancock and Warm (1989). They distinguished three facets of stress and labeled them the "trinity of stress," as shown in Figure 4.2. Input refers to the environmental events to which an individual is exposed, which include information (i.e., displays) as well as traditional input categories such as temperature, noise, and vibration (e.g., Hancock, Ross, and Szalma 2007; Pilcher et al. 2002). The second is adaptation, which encompasses the appraisal mechanisms referred to previously. The third and final component is output level, which indicates how an organism behaves with respect to goal achievement. A fundamental tenet of the Hancock and

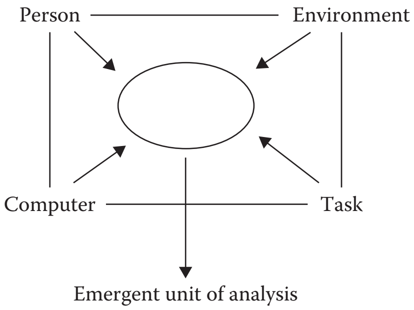


FIGURE 4.1 An illustration of the emergence of a supraordinate unit of analysis that derives from the interaction of an individual (person), the tool he or she uses (computer), the task he or she has to perform (task), and the context (environment) against which the action occurs.

Warm (1989) model is that in a large majority of situations (and even in situations of quite high demand) individuals do adapt effectively to input disturbance. That is, they can tolerate high levels of either overload or underload without enormous change to their performance capacity. Adaptive processes occur at multiple levels, some being the physiological, behavioral (i.e., performance), and subjective/affective levels. These adaptations are represented in the model as a series of nested, extended inverted-U functions (see Figure 4.3) that reflect the fact that under most conditions the adaptive state of the organism is stable. However, under extremes of environmental underload or overload, "failures" in adaptation do occur. Thus, as the individual is perturbed by the input, the first threshold one traverses is subjective comfort. This is followed by behavioral effects and finally failure of the physiological system (e.g., loss of consciousness). Examples of such extreme failures are relatively rare in most settings, although when they do occur (e.g., in conflict situations) they are often catastrophic for the individual and the system he or she is operating (e.g., Harris, Hancock, and Harris 2005).

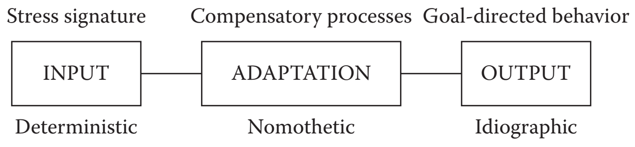


FIGURE 4.2 The trinity of stress, which identifies three possible "loci" of stress. It can be viewed as an input from the physical environment, which can be described deterministically. Since such a profile is by definition unique, it is referred to as a stress signature. The second locus is adaptation, which describes the populational or nomothetic reaction to the input itself. It is most evidently measurable in the processes of compensation. The third and final locus is output, which is expressed as the impact on the ongoing stream of behavior. Since the goals of different individuals almost always vary, this output is largely idiographic or person specific. It is this facet of stress that has been very much neglected in prior and contemporary research.

This model is unique in that it provides explicit recognition that the *proximal form of stress* in almost all circumstances is the task itself. Task characteristics are incorporated in the model by two distinct base axes representing spatial and temporal components of any specified task. Information structure (the spatial dimension) represents how task elements are organized, including challenges to such psychological capacities such as working memory, attention, decision making, and response capacity. The temporal dimension is represented by information rate. Together, these dimensions can be used to form a vector (see Figure 4.4) that serves to identify the current state of adaptation of the individual. Thus, if the combination of task characteristics and an individual's stress level can be specified, a vector representation can be used to predict behavioral and physiological adaptation. The challenge lies in quantifying the information-processing components of cognitive work (see Hancock, Szalma, and Oron-Gilad 2005).

Although the model shown in Figure 4.4 describes the level of adaptive function, it does not articulate the mechanisms by which such adaptation occurs. Hancock and Warm (1989) argued that one way in which individuals adapt to stress is by narrowing their attention by excluding task-irrelevant cues (Easterbrook 1959). Such effects are known to occur in

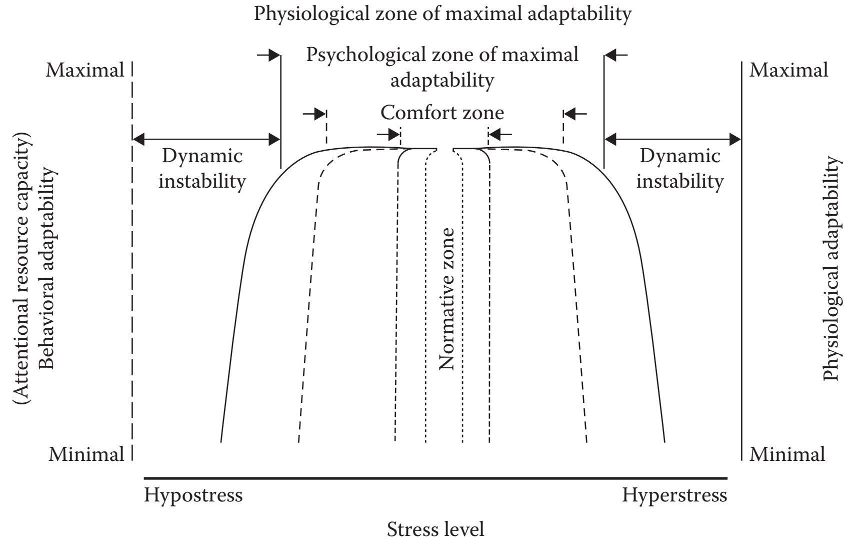


FIGURE 4.3 The extended-U relationship between stress level and response capacity. As is evident, the form of degradation is common across the different reflections of response. At the center of the continuum is the normative zone, which reflects optimal functioning. Outside of this is the comfort zone, which reflects the behavioral recognition of a state of satisfaction. Beyond this lies the reaction of psychological or cognitive performance capacity. Finally, the outer envelope is composed of physiological functioning. There are proposed strong linkages between the deviation from stability at one level being matched to the onset of radical failure at the more vulnerable level that is nested within it. The model is symmetrical in that underload (hypostress) has mirror effects to overload (hyperstress), which is usually considered the commonly perceived interpretation of stress.

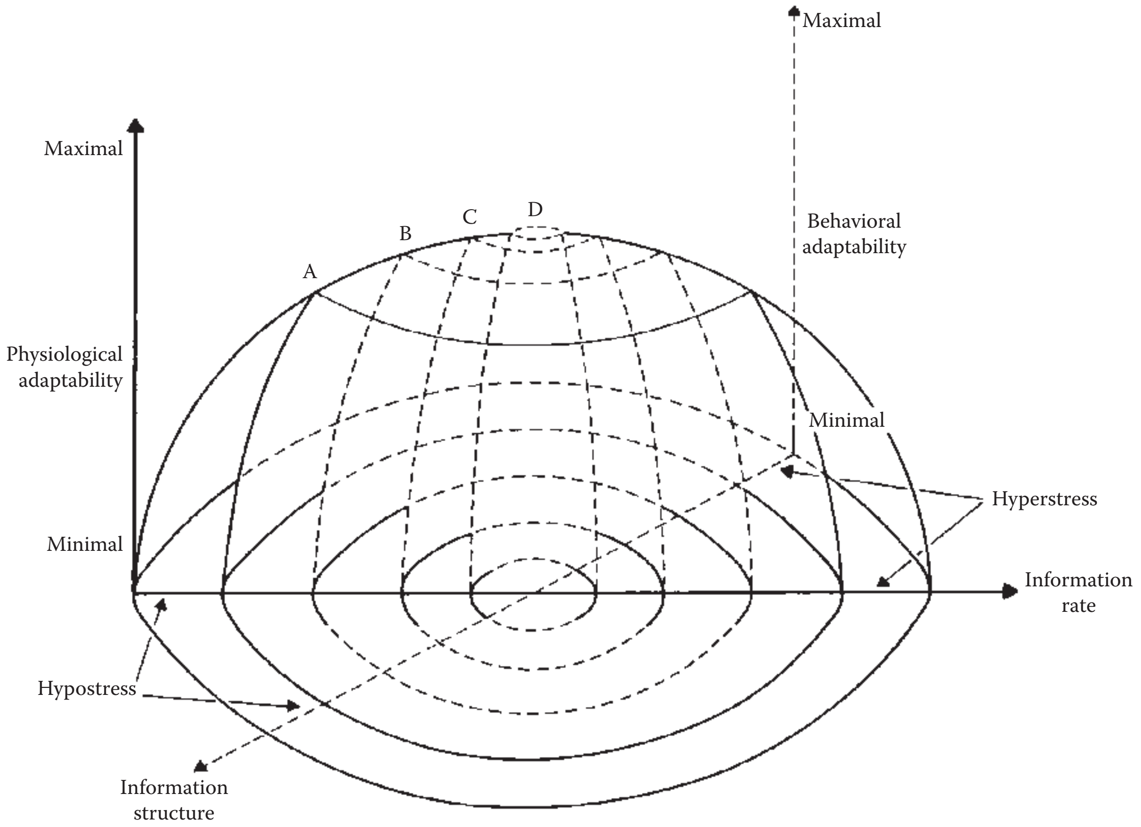


FIGURE 4.4 A three-dimensional representation of Figure 4.3. The description given in Figure 4.3 is now expanded into a three-dimensional representation by parsing the base hypostress–hyperstress axis into its two component elements. These divisions are composed of information rate (the temporal axis) and information structure (the spatial axis). Note that any one source of input stress can be described as a scalar on the base axis and these scalars can be summed to provide a multi-input stress vector that then provides a prediction of both performance and physiological adaptability, which are the primary descriptors on the vertical axis. A, B, C and D represent the thresholds of adaptability. A represents the physiological zone of maximal adaptability; B is the psychological zone of adaptability, C is the comfort zone, while D illustrates the normative zone.

spatial perception (e.g., Bursill 1958; Cornsweet 1969), and narrowing can occur at the levels of both central and peripheral neural systems (Dirkin and Hancock 1984, 1985; Hancock and Dirkin 1983). Further, Hancock and Weaver (2005) have argued that distortions of temporal perception under stress are related to this narrowing effect. However, evidence suggests that these two perceptual dimensions (space and time) may not share common perceptual mechanisms (see Ross et al. 2003; Thropp, Szalma, and Hancock 2004).

4.2.3 The Cognitive–Energetic Framework

The Hancock and Warm model accounts for the levels of adaptation and adaptation changes that occur under the driving forces of stress. However, it does not articulate precisely how effort is allocated under stress or the mechanisms by which individuals appraise the task parameters that are the proximal source of stress. The precise effort allocation issue is addressed by a cognitive–energetic framework described by Hockey (1997). The compensatory control model is based on three assumptions: (1) Behavior is goal directed. (2) Self-regulatory processes control goal states. (3) Regulatory activity has energetic costs (i.e., it consumes resources). In this model, a compensatory control mechanism allocates resources dynamically according to the goals of an individual and the environmental constraints. The mechanisms operate at two levels (see Figure 4.5): The lower level is more or less "automatic" and represents established skills. Regulation at this level requires few energetic resources or

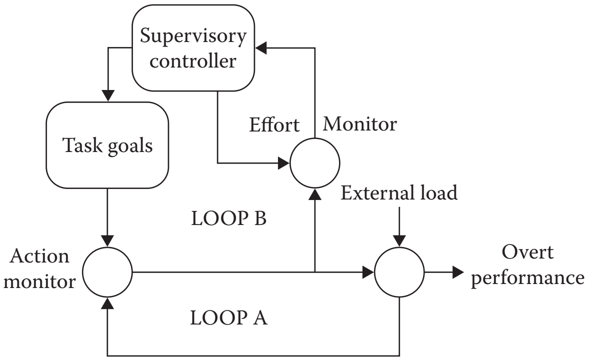


FIGURE 4.5 The two-level effort regulation model by Hockey: This model provides a mechanism by which an individual allocates limited cognitive resources to different aspects of performance. (From Hockey, G. R. J., *Biol Psychol*, 45:73–93, 1997. With permission.)

low active regulation and effort (cf. Schneider and Shiffrin 1977). The upper level is a supervisory controller, which can shift resources (effort) strategically to maintain adaptation, and reflects effortful and controlled processing. The operation of the automatic lower loop is regulated by an effort monitor, which detects changes in the regulatory demands placed on the lower loop. When demand increases beyond the capacity of the lower loop, control is shifted to the higher, controlled processing loop. Two strategic responses of the supervisory system are increase in effort and change in the goals. Goals can be modified in terms of their kind (change the goal itself) or strength (e.g., lowering the criterion for performance). From a self-regulation perspective, these modifications adjust the discrepancy between goal state and the current state by increasing effort or changing the goal (see Carver and Scheier 1998).

4.3 RELATIONSHIP BETWEEN STRESS AND COGNITIVE WORKLOAD

4.3.1 Cognitive Workload as a Form of Stress

The Hancock and Warm (1989) model explicitly identifies the task itself as the proximal source of stress. In operational environments, this is often manifested as increases or decreases in cognitive workload (Moray 1979). As in the case of stress, workload is easily identified colloquially but difficult to define operationally. Workload can manifest in terms of the amount of information to be processed (an aspect of information structure) and the time available for processing (information rate). Thus, the base axes of the Hancock and Warm model capture dimensions of workload as well as stress (and see Hancock and Caird 1993). Indeed, physiological measures of workload (O'Donnell and Eggemeier 1986) are often the same as the measures used to assess physiological stress. Similarly, subjective measures of workload and stress both reflect appraisals of the task environment and of its perceived effect on the individual (Hart and Staveland 1988). Although the two concepts were developed in separate research traditions, the artificial boundary between them should be dissolved as each term refers to similar processes. The implication for HCI is that computer-based tasks that impose either too much or too little demand will likely be appraised as stressful. In the latter case, the underload stress will be interpreted as boredom. Thus, the design process for the development of computer interfaces should include assessment of perceived workload as well as affective state.

4.3.2 Performance and Workload: Associations/Dissociations

It is often the case that performance is maintained under increased workload/stress, which is reflected in the extended-U model described by Hancock and Warm and in the mechanisms of Hockey's energetic model of compensatory control. Maintaining performance under stress is associated with costs, both physiologically and cognitively. Further, one would expect that in easier tasks performance is not as costly and that, therefore, there is a direct *association* between task difficulty and perceived workload. Such performance–workload associations do occur, and they occur most prevalently in vigilance tasks (Warm, Dember, and Hancock 1996; see also Szalma et al. 2004). However, other forms of workload–performance relations can occur. For instance, perceived workload may change as a function of change in task demand, but performance remains constant. Hancock (1996) refers to these situations as *insensitivities*, which can be diagnostic with respect to the relation between the individual and the task (see also Parasuraman and Hancock 2001). Thus, consistent with the frameworks of Hancock and Warm (1989) and Hockey (1997), one response to increased task demand is to exert more effort, thereby maintaining performance but increasing perceived workload. Alternatively, one could have a situation in which task demands increase and performance decreases, but perceived workload does not change. This suggests that appraisals of a task are not always sensitive to actual changes in that task.

Interesting corollaries of these observations are performance–workload *dissociations* that sometimes occur (Hancock 1996; Yeh and Wickens 1988). In such cases, decreased performance is accompanied by *decreased* workload. One possible reason for such a result is disengagement of the individual from the task (i.e., the person gives up; see Hancock 1996). In the case where increased performance is observed to be accompanied by increased perceived workload, the pattern suggests effective improvement of performance at the cost of increased effort allocation. An area of much-needed research is establishing which task parameters control the patterns of performance–workload associations and dissociations, and how these change dynamically as a function of time on task. It may well be that reformulating the task by innovations in the interface itself addresses these crucial concerns (see Hancock 1997). Indeed, the structure and organization of computer interfaces will be a major factor in determining both performance under stress and the relation of performance to perceived workload.

4.4 MITIGATION OF STRESS

If changing the fundamental nature of demand is one solution, we now look at other approaches to mitigate the negative effects of stress and workload on the performance of HCI tasks. These strategies include skill development (e.g., Hancock 1986), specific display design changes (Hancock and Szalma 2003a; Wickens 1996), as well as technologies employing adaptive automation and decision aids (Hancock and Chignell 1987). Developing skills so that they are relatively automatic as opposed to the alternative controlled processing (Schneider and Shiffrin 1977) and developing expertise can mitigate some of the negative effects of stress (Hancock and Hancock 2010). Regarding display design, simple, easily perceivable graphics can permit quick, direct extraction of information when cognitive resources are reduced by stress and workload (Hancock and Szalma

2003a). Adaptive automation can be employed by adjusting the level of automation and the management of that automation according to stress state (e.g., Scerbo, Freeman, and Mikulka 2003). In addition, adapting the form of automation (i.e., level, management type) to the operator based on their own personal style of interaction can serve to improve its utility for aiding performance and reducing stress and workload (see Thropp et al. 2004). Indeed, experimental findings are even now beginning to establish the relation for automation to effectively mitigate performance-related stress (Funke et al. 2007).

4.4.1 Theoretical Bases of Emotion (Stress) Regulation

In both theory and research, stress is clearly linked with the more general topic of emotion (Lazarus 1999). Indeed there is growing recognition of the need to consider a user's emotional response to a task or an interface as it is an important aspect of design. Emotions are valenced reactions to either internal or external stimuli, which trigger multisystemic changes in both physiology and behavior (Ochsner and Gross 2005). Emotions are therefore useful in presenting feedback concerning an operator's ongoing interaction with the environment and especially the computer-based technology with which they must interact (Folkman et al. 1986). The computer and the manner in which it functions can produce a range of emotional reactions in the operator who is attempting to manipulate the system. However, at present the computer has no concept of emotional experience or display, no matter how often humans might attribute these characteristics to the machine (Luczak, Roetting, and Schmidt 2003). Contrary to intuition, the computer is not malfunctioning in order to frustrate or spite its users and no amount of shouting or banging will presently instill it with a sense of motivation to work. However, this is not to say extensive efforts are not underway to develop computer systems that both recognize user emotions and generate emotional expressions on behalf of the computer system (Zhang et al. 2010). The very act of interacting with a computer represents an emotional experience as the machine is a tool by which the operator hopes to accomplish his or her desired goals. Events whereby the computer facilitates the achievement of a goal can result in pleasant emotions (i.e., accomplishment, relief, and happiness), whereas instances in which the interface component of the human–computer dyad is perceived as detrimental or as a barrier to goal fulfillment can produce negative valence emotions such as anger and frustration (Hassenzahl and Ullrich 2007). Therefore, emotions are an inherent feature of HCI as no human action, even one performed in tandem with an affectless instrument, takes place in a completely emotion-free context. Emotions then have the potential in HCI to become either stressors themselves or tools by which operators can cope with stress and enhance the effectiveness and efficiency of performance. Gross, Richards, and John (2006) have postulated that effective emotion regulation is a qualification for adaptive functioning in almost all everyday skills. Given the growing popularity and availability of technologies such as personal digital assistants and cellular telephones, HCIs are rapidly becoming modal everyday tasks. Techniques for regulating the pervasive influence of emotions are therefore useful skills for the operator to develop so that they may minimize the disadvantageous consequences of negative emotional experiences.

4.4.1.1 Psychological and Physiological Strategies for Emotion Regulation

Emotions are typically categorized based on two componential characteristics: (1) valence and (2) arousal (Lang 1995). Valence refers to the extent to which an operator interprets an emotion as pleasant or unpleasant. Arousal constitutes the extent to which an emotion evokes a response from the operator's physiological system. Stress is unique in that it has the ability to run the gamut on both dimensions; thus, it can be perceived as both pleasant and unpleasant, as well as inducing either mild or severe physiological reactions. Techniques for its regulation therefore incorporate methods to address both the psychological and physiological components of stress (see Hancock and Warm 1989).

A typical course of emotional experience, without any attempts at emotion regulation, begins with emotionally charged environmental cues eliciting intrapersonal emotional response tendencies, which lead to emotional responses (Gross 1998). Emotion regulation techniques may therefore intercede at a number of points in this process. Antecedent-focused strategies, such as cognitive reappraisal, specifically endeavor to manipulate the interaction between emotional cues and their subsequent response tendencies and are therefore a method of evaluation (i.e., viewing the interaction with the computer as a learning experience). Response-focused strategies such as expressive suppression, on the other hand, affect the relationship between an operator's response tendencies and the resultant emotional response and are therefore a method of modulation (i.e., ignoring any unpleasant feelings resulting from interaction with the computer). Both techniques have proved to be effective strategies, although the optimal technique heavily depends on the situation (Gross 2002).

4.4.1.2 Deliberate Emotion Regulation

Emotion regulation entails "processes that individuals use to influence which emotions they generate, when they do so, and how these emotions are experienced or expressed" (Ochsner and Gross 2005, p. 243). Active processing of this nature, which requires attentional resources, is referred to as deliberate emotion regulation. Recent research suggests the possibility that emotion regulation can take place automatically, at an unconscious level (Mauss, Bunge, and Gross 2007). While seminal research efforts investigating the influence of automatic emotion regulation on performance are currently underway (Hancock and Beatty 2010), our focus here necessarily concentrates on the more established deliberate emotion regulation strategies. The two strategies that are studied most often are cognitive reappraisal and expressive suppression. These techniques share a common goal of emotion regulation, but they differ in the aspects of emotion that they influence, when they begin to influence emotional experience, and their long-term versus short-term effectiveness.

4.4.1.3 Cognitive Reappraisal

Cognitive reappraisal is defined as the act of interpreting potential emotion-provoking stimuli in unemotional terms (Speisman et al. 1964). The purpose of cognitive reappraisal is therefore to influence an individual's cognition, in order to maintain control over emotional responses. As mentioned in Section 4.4.1.1, cognitive reappraisal is an antecedent-focused emotional regulation strategy. The intervention occurs as early as possible in the emotional experience so as to minimize deleterious performance consequences. Indeed, Gross (2002) reported that reappraisal is more effective than expressive suppression. Unlike suppression, it has no detrimental effects on other cognitive processes such as memory. The utilization of cognitive reappraisal is also associated with superior long-term health outcomes (Haga, Kraft, and Corby 2009).

4.4.1.4 Expressive Suppression

Expressive suppression is defined as the inhibition of emotionally expressive behavior despite emotional arousal (Gross and Levenson 1993). The primary aim of the suppression approach is therefore to minimize outward displays of emotion, that is, targeting overt behavior instead of cognition. As a response-focused strategy, suppression initiates its influence later in the process sequence than cognitive reappraisal (Gross 1998). Although cognitive reappraisal is more effective over the long term, suppression is the superior short-term option; operators employing expressive suppression may have more attentional resources available as they are not actively engaging in the continual assessment and reassessment of environmental stimuli. Operators engaging in HCIs under time constraints may therefore find this strategy more effective.

4.4.2 Changing the Person

To improve a stressful human-machine interaction, one viable option is to alter the human's attitudes or abilities either through training or selection.

4.4.2.1 Training/Skill Development

Clearly, the greater the skill of an individual the more resilient his or her performance under stress (Hancock 1986). This well-established phenomenon is incorporated into the energetic theories of stress and performance and is an approach most often taken to mitigate adverse workload and stress effects. However, training on relevant tasks is only one method of training for stress. There are also techniques for training individuals to cope more effectively with stress itself, such as the aforementioned emotion regulation techniques discussed in Sections 4.4.1.3 and 4.4.1.4, which essentially build stress-coping skills. An additional example of such an approach is stress exposure training (SET; Johnston and Cannon-Bowers 1996), a three-phase procedure in which individuals are provided information regarding the stress associated with task performance, are provided training on the task, and then practice their task skills under simulated stress conditions. This technique has been shown to be effective in reducing anxiety and enhancing performance (Saunders et al. 1996). There is evidence that coping skills learned with a particular type of stressor and task can be transferred to novel stressors and tasks (Driskell, Johnston, and Salas 2001). For such an intervention to succeed, however, it is crucial that the training is designed based on a complete and accurate analysis of the task environment (Johnston and Cannon-Bowers 1996). If task parameters that are the most responsible for the workload and stress are identified, these can be especially targeted in training.

An additional issue in training for more effective stress coping is modifying an individual's appraisal of events, an approach that is coincident with the emotion regulation technique of cognitive reappraisal discussed in Section 4.4.1.3. Learning to approach HCIs with effective coping is therefore a valuable skill to acquire as early as possible in any task training. Inducing automaticity in some skills permits reallocation of resources to coping efforts, as well as reducing the likelihood that the task environment itself is appraised as threatening. Even if an event is appraised as a threat to an individual's psychological or physical well-being, the highly skilled individual will appraise his or her coping ability as sufficient to handle such an increased demand. However, there has been limited research on how individuals who develop expertise also develop the capacity to effectively cope with the stress that accompanies performance in a given domain and on the extent to which stress-coping skills in one domain transfer to other domains. Deliberate practice generally facilitates skill development (Ericsson 2006). If one considers coping with stress to be a skill, then in principle deliberate practice should permit the development of expertise in coping with stress. This likely involves parsing the task into components, based on cognitive task analysis, and designing training procedures that target the stressful aspects of the task. However, such efforts require an understanding of how different forms of stress affect different forms of information processing. Since these variables are difficult to quantify, establishing these linkages must be driven by theory. Elucidation of these issues will provide the groundwork for future development of stress mitigation tools during training and skill development.

4.4.2.2 Personnel Selection

Selection techniques have been a popular choice for matching individuals to specific jobs, but the focus has traditionally and historically been on intellectual skills (e.g., Yerkes 1918). Selecting individuals for their stress-coping capability has been applied to the selection criteria for police officers, who therefore tend to be as stable as or more emotionally stable than the rest of the population (for a review, see the work by Brown and Campbell [1994]). Selecting individuals with proficient stress-coping skills becomes still more difficult given the complex criteria that define stress and the fact that "successful" coping skills vary by situation and desired outcome. Research is therefore needed that links particular traits to stress-coping skills for specific task environments. The effectiveness of everyday life stress coping, such as that observed in individuals who are extraverted (McCrae and Costa 1986; Penley and Tomaka 2002) or optimistic (Aspinwall, Richter, and Hoffman 2002; Scheier and Carver 1985), may not predict effective coping in specific task domains. Understanding which individuals will likely cope effectively with a particular task therefore requires first a thorough understanding of the perceptual, cognitive, and psychomotor characteristics of the task and then linking these parameters to trait profiles. By far the most research on the relation of affective traits to task performance has been conducted in the areas of extraversion and trait anxiety/neuroticism (see the work by Matthews, Deary, and Whiteman [2003] for a review). However, the characteristics of greatest interest may vary somewhat across domains, although some general traits (e.g., emotional stability, conscientiousness) would be expected to moderate performance across a variety of task environments.

4.4.3 Changing the Task

Modifying the technology component is another possibility for more effective HCI.

4.4.3.1 Display Design

Although training and selection can mitigate stress effects, the primary method of stress mitigation requires the tasks themselves to be redesigned. This is for two reasons: (1) There will be many instances where selection is not possible and expenditure of significant resources on training is undesirable. (2) There are instances in which one wishes to design an interface that requires little or no training and that can be used by any member of a large population of individuals (e.g., consumers). Particularly in light of the observation that task represents the proximal source of stress, future work in stress mitigation for HCI should focus on redesign of the task and of the interface itself. In previous work, we have argued that existing display design techniques that are simple and easily perceived would be the best choice for an interface that is used in stressful environments (Hancock and Szalma 2003a). Specifically, configural or object displays can represent complex, multivariable systems as simple geometric shapes or emergent features if those features are mapped well to system dynamics (see Bennett and Flach 1992). Under stress, it is the complex problem solving and analytical skills that are the most vulnerable and are apt to decline first. A display that allows fast extraction of information with minimal cost in working memory load can mitigate stress effects (Hancock and Szalma 2003a; Wickens 1996). A combination of training to automaticity and displays of information that can be perceived directly with a minimum of information-processing requirements is currently one of the best approaches for stress mitigation in cognitively complex environments.

4.4.3.2 Adaptive Automation

Another approach for stress mitigation is the allocation of function to automated systems (Hancock and Chignell 1987). The advent of modern automated systems allows for automation to adapt to the state of an individual (Scerbo, Freeman, and Mikulka 2003). Thus, at points in time when an operator is overstressed and overtaxed, the system can assume control of some task functions, thereby freeing resources to effectively cope with increased task demand. Two potential problems for automated systems are that overreliance can occur and operator skills can atrophy. However, a dynamic (adaptive) automated system that permits or requires the operator to perform functions at different points in time can reduce the probability of skill atrophy while still relieving the workload and stress of task performance.

However, the introduction of automation can itself induce stress. Operators who work with automated systems, particularly static, inflexible automated systems, are relegated to the role of monitors who must respond only when untoward events occur. Sustained attention requirements are in fact quite stressful (Warm, Parasuraman, and Matthews 2008) and paradoxically induce high perceived workload (Warm, Dember, and Hancock 1996). Adaptive automation can mitigate this problem by dynamically assigning tasks to the machine or the human being depending on the environmental conditions and the state of the operator (Hancock and Chignell 1987). Indeed, potential techniques for enabling the operator's neurological state to adjust to automation have been identified (e.g., Scerbo, Freeman, and Mikulka 2003).

4.4.4 Hedonomics: Promoting Enjoyable Human–Computer Interaction

Stress research has traditionally followed the edict of ergonomics and human factors, in general, to first do no harm and then seek to prevent pain and injury. As with the rest of behavioral science, stress researchers have often sought to treat the symptoms of stress and mitigate its negative effects on performance. However, with the advent of positive psychology (Seligman and Csikszentmihalyi 2000), there has been a movement to incorporate the promotion of pleasure and well-being rather than restrict efforts to pain prevention. Hancock (Hancock, Pepe, and Murphy 2005) coined the term hedonomics and defined it as that branch of science that facilitates the pleasant or enjoyable aspects of human–technology interaction. In short, the goal of hedonomics is to design with happiness in mind. Hedonomics is a fairly new research area, but during the last decade there has been a rapid growth in research concerning affect and pleasure. Affective evaluations provide a new and different perspective in human factors engineering. It is not how to evaluate users; it is how the user evaluates (Hancock et al. 2005). The research on hedonic values and seductive interfaces is in fact a welcome contrast to research on safety and productivity, which have dominated human factors and ergonomics (HF/E) for so long. It must be noted, however, that *pleasurable* interaction with technology is not necessarily conducive to *happiness*. Indulging in pleasures can sometimes interfere with happiness and well-being (see Fromm 1976; Kasser 2002; Ryan and Deci 2001).

Our argument is not that we should discard current methods in HF/E. Clearly functionality and usability are necessary conditions for pleasurable interaction with technology. If an interface does not function in a way that is congruent with the user's goals so that the user appraises the technology as an agent that is interfering with goal achievement, that interaction is likely to be stressful and performance may decline. However, functionality and usability are necessary but not sufficient conditions for pleasurable interactions with technology. The interface should be designed such that it affords appraisals of the technology as a convivial tool (Illich 1973) or aid. One can also utilize the human tendency to anthropomorphize technology to facilitate such appraisals of the technology as "helpful and supportive" rather than as "conflictive" or, worse, an "enemy" (Luczak, Roetting, and Schmidt 2003).

Hedonomic design is of obvious importance for the development of consumer products, but in principle it can also transform the very nature of work itself, rendering it "fun." Although there may be some tasks that will never be completely enjoyable, there are many individuals who have jobs that could be made more enjoyable by designing the tasks such that they promote teletic work (Csikszentmihalyi 1990) while also facilitating intrinsic motivation (Deci and Ryan 2000).

4.4.4.1 Teletic Work and Intrinsic Motivation

A useful theoretical framework for hedonomics is self-determination theory (SDT; Deci and Ryan 1985, 2000; Ryan and Deci 2000, 2001). From this perspective, there are three organismic needs that are essential for facilitating intrinsic motivation for task activity and the positive affect that can accompany such states. These needs are competence (self-efficacy; see also Bandura 1997), autonomy (personal agency, not independence per se), and relatedness. An important difference between this theory and other theories of motivation is the recognition that there are qualitatively different forms of motivation (Gagne and Deci 2005). Thus, in SDT five categories of motivated behavior are identified that vary in the degree to which motivation is self-determined. Four of the categories reflect extrinsic motivation and one category is intrinsic motivation. In the latter case, individuals are inherently motivated to engage in activity for its own sake or for novelty and challenge. The four extrinsic motivation categories vary in the degree to which regulation of behavior is internalized by the individual and, therefore, they are more autonomous and self-determined (Ryan and Deci 2000). The process of internalization involves transforming an external regulation or value into one that matches an individual's own values. The development of such autonomous motivation is crucial to skill development, since the person must maintain his or her effort throughout a long and arduous process. Individuals who are autonomously motivated to learn are those who develop a variety of effective self-regulation strategies, have high self-efficacy, and set a number of goals for themselves (Zimmerman 2000). Further, effective self-regulation develops in four stages: (1) observation, (2) emulation, (3) self-control, and (4) self-regulation. Successful skill development involves focus on process goals in the early stages of learning and outcome goals in the fourth stage (Zimmerman 2000).

4.4.4.2 Intrinsic Motivation and Skill Development

Research has established that intrinsic motivation is facilitated by conditions promoting autonomy, competence, and relatedness (see Deci and Ryan 2000). Three factors that support autonomy are as follows: (1) meaningful rationales for doing a task, (2) acknowledgment that the task might not be interesting, and (3) an emphasis on choice rather than control. It is important to note that externally regulated motivation predicts poorer performance on heuristic tasks (Gagne and Deci 2005), suggesting that as experts develop better knowledge representations it will be crucial to promote their internal regulation of motivation. Although intrinsic motivation has been linked to how task activities and environmental contexts meet psychological needs, it is not clear why skilled performers are able to meet these needs or why an individual chooses a particular computer interface. It is likely that interest in activities codevelops with abilities and traits (see Ackerman and Heggestad 1997), but this issue needs more thorough investigation in the context of complex computer environments that require highly skilled work.

Emotions can be powerful motivators of task performance, including tasks involving HCI. Both intrinsic motivation and emotional experience play critical roles in beginners' perceptions concerning their current and future interactions with a system (Venkatesh 2000). Although emotion cannot and most probably should not be designed out of an HCI, it is possible to design activities in which emotional experience facilitates learning and enhances an operator's intrinsic motivation for task mastery (Lepper and Cordova 1992). Such activities should also aim to simultaneously foster effective emotion regulation techniques. Learning to perform in the presence of common stressors early in the learning process will help to maintain task engagement. In addition to the aforementioned concern and the issues of efficacy and self-regulation, there is a need to examine the process by which individuals internalize extrinsic motivation as gain experience with a particular interface or system. In particular, Gagne and Deci (2005) noted that little research has examined the effect of reward structures and work environments on the internalization process. It is likely that environments structured to meet basic needs more likely facilitate internalization processes and inoculate learners against the trials and tribulations that face them as they interact with new technologies.

4.4.4.3 Teletic Work and Motivational Affordances

Teletic, or autoteletic, work refers to work that is experienced as enjoyable and is associated with "flow" or optimal experience characterized by a sense of well-being and harmony with one's surroundings (Csikszentmihalyi 1990). There is variation in both tasks and individuals with respect to the degree to which the human–technology interaction is teletic. There are four categories in which individuals tend to fall with respect to their relation to work: First, there is a small proportion of the population that is always happy in life, regardless of their activity. Csikszentmihalyi (1990) refers to these individuals as having an "autotelic personality." There is also a group of individuals who are naturally predisposed to being happy regarding a specific task. They appraise such tasks as enjoyable and often seek out these activities. The third group consists of individuals who enjoy specific activities but cannot do them professionally. This group includes individuals such as amateur athletes. The vast majority of people, however, do work for purely functional reasons (e.g., finances and security). For these individuals, work is boring and grinding because the task itself is nearly always considered aversive. A goal of hedonomics is to design work that can be enjoyed to the greatest extent possible. This means structuring the environment as an entire system, ranging from the specific cognitive and psychomotor demands to the organization in which a person works. Even in jobs that are not inherently enjoyable, some degree of positive affect can be experienced by workers if their environment is structured to facilitate a sense of autonomy (personal agency), competence, and relatedness (Deci and Ryan 2000; see also Gagne and Deci 2005). From an ecological perspective (Flach et al. 1995), this means identifying the *motivational affordances* in the task and work environment, and designing for these affordances. Thus, just as one might analyze the affordance structure of an interface using ecological interface design methods (e.g., Vicente and Rasmussen 1992), one can design an environment so that the elements of the physical and social environment afford stress reduction and enhanced intrinsic motivation. An affordance is a relational property that does not exist independent of the individual and the environment. Affordances therefore share conceptual elements of person–environment transactions that drive emotion and stress. They differ from each other in that the classical definition of affordance often describes it as a physical property of the environment (Gibson 1966, 1979), although more recent thinking suggests that no specific physical element connotes affordance. Thus, one cannot define either concept by isolating either the individual or the context (see Reed 1996).

Motivational affordances may be conceived as elements of the work environment that facilitate and nurture intrinsic motivation. The key for design is to identify *motivational invariants*, or environmental factors that consistently determine an individual's level of intrinsic motivation across contexts. There are some aspects of work that have been identified as important for facilitating intrinsic motivation and would thus be considered motivational invariants. For instance, providing feedback that is perceived as controlling rather than informative tends to undermine a sense of autonomy and competence and thereby reduces intrinsic motivation (Deci, Ryan, and Koestner 1999). Careful analyses of the motivational affordance structure permit design of tasks that are more likely to be enjoyable by rendering the tools convivial (Illich 1973) and thereby facilitating human–machine synergy (see Hancock 1997).

4.5 DIRECTIONS FOR FUTURE HUMAN–COMPUTER INTERACTION RESEARCH IN RELATION TO WORKLOAD AND STRESS

In this section, we identify directions for future research. These include a better understanding of resources and quantifying task dimensions defined in the Hancock and Warm (1989) model. Progress here likely reduces to the thorny problem of quantifying human information processing (see Hancock, Szalma, and Oron-Gilad 2005). Further, we discuss the need for research on performance–workload associations and dissociations, and the evident need for programmatic investigation of the role of individual differences in performance, workload, and stress.

The Hancock and Warm (1989) model of stress explicitly identifies task dimensions that influence stress state and behavioral adaptability. However, the metrics for these dimensions, and how specific task characteristics map to them, have yet to be fully articulated. Thus, future research should aim to examine how different task components relate to performance and subjective and physiological states. Development of a quantitative model of task characteristics will permit the derivation of vectors for the prediction of adaptability under stress. Cognitive neuroscience and neuroergonomics in particular offer a very promising approach to such understanding. An additional step in this direction, however, will be facilitated by improved quantitative models of how human beings process information (Hancock, Szalma, and Oron-Gilad 2005).

4.5.1 Understanding Mental Resources

One of the challenges for quantifying human information processing is that there is little understanding or consensus regarding the capacities that "process" the information. A central concept in energetic models of human performance is mental resources. Resource theory replaced arousal and serves as an intervening variable to explain the relations between task demand and performance. However, a continual problem for the resource concept is to operationally define what resources actually are. Most early treatments of resources used that term metaphorically (Navon and Gopher 1979; Wickens 1980, 1984), and the failure to specify what resources have led some to challenge the utility of the concept (Navon 1984). As resource theory is a central concept in theories of stress and represents one of the most important issues to be resolved in future research on stress and performance, we now turn to the definitional concerns associated with the resource construct and imperatives for future research to refine the concept.

4.5.1.1 Resource Metaphors

Two general categories of resource metaphors may be identified as structural metaphors and energetic metaphors. One of the earliest conceptualizations of resource capacity used a computer-based metaphor (Moray 1967). Thus, cognitive capacity was viewed as being analogous to the random access memory and processing chip of a computer, consisting of information-processing "units" that can be deployed for task performance. However, the structural metaphor has been applied more to theories of working memory than to attention and resource theory.* Most early resource theories, including Kahneman's (1973) original view and modifications by Norman and Bobrow (1975), Navon and Gopher (1979), and Wickens (1980, 1984), applied energetic metaphors to resources. These perspectives conceptualized resources as commodities or as pools of energy to be "spent" on task performance. In general, energetic approaches tend to employ either economic or thermodynamic/hydraulic metaphors. The economic model is reflected in the description of resources in terms of supply and demand: Performance on one or more tasks suffers when resource demands of the tasks exceed available supply. Presumably the total amount of this supply fluctuates with the state of the individual, and the "assets" diminish with increases in the intensity or duration of stress. Although Kahneman's (1973) original conception allows for dynamic variation of available resource capacity, most early models assumed a fixed amount of resources (see Navon and Gopher 1979). In thermodynamic analogies, resources comprise a fuel that is consumed or a tank of liquid to be divided among several tasks, and under stressful conditions the amount of resources available is insufficient to meet demand and thus performance suffers. There is no consensus as to the capacity or flexibility of such a tank or whether there are numerous malleable tanks that only store modality-specific resources (Young and Stanton 2002). In discussing his version of resource theory, Wickens (1984) warned that the hydraulic metaphor should not be taken too literally, but most subsequent descriptions of resources have employed visual representations of resources in just this form (i.e., a tank of liquid). Similarly, many discussions of resource availability and expenditure adopt the economic language of supply and demand, and Navon and Gopher (1979) explicitly adopted principles of microeconomics in developing their approach. An additional problem faced by resource theory is that in most cases (e.g., Navon and Gopher 1979; Wickens 1980, 1984) the structural and energetic metaphors are treated interchangeably, a further testament to the ambiguity of the construct.

A problem with using nonbiological metaphors to represent biological systems is that such models often fail to capture the complexity and the unique dynamic characteristics (e.g., adaptive responses) of living systems. For instance, a hydraulic model of resources links the activity of a tank of liquid, governed by thermodynamic principles, to the action of arousal mechanisms or energy reserves that are allocated for task performance. However, a thermodynamic description of the physiological processes underlying resources is at a level of explanation that may not adequately describe the psychological processes that govern performance. Thermodynamic principles can be applied to the chemical processes that occur within and between neurons, but they may be less useful in describing the behavior of large networks of neurons.† Similarly, economic metaphors of supply and demand may not adequately capture the relation between cognitive architecture and energy allocated for their function. Economic models of resources define them as commodities to be spent on one or more activities and they assume an isomorphism between human cognitive activity and economic activity, an assumption that may not be tenable. Indeed, Navon and Gopher (1979) admitted that their "static" economic metaphor for multiple resources may need to be replaced by a dynamic one that includes temporal factors (e.g., serial versus parallel processing; activity of one processing unit being contingent on the output of another). Such concerns over the metaphors used to describe resources are hardly new (Navon 1984; Wickens 1984); but the use of metaphors has become so ingrained in the general scientific thinking about resources and human performance that reevaluation of metaphors is more than warranted, it should be mandated. A regulatory model based on neurophysiological chemistry may serve as a better metaphor (and, in future, may serve to describe the actual nature of the resources themselves to the extent that they can be established) to describe the role of resources in human cognition and performance. However, even a physiology-based theory of resources must be tempered by the problems inherent in reducing psychological processes to physiological activity.

4.5.1.2 Function of Resources

Another problem faced by resource theory is the absence of a precise description of how resources control different forms of information processing. Do resources determine the energy allocated to an information processor (Kahneman 1973), do they provide the space within which the processing structure works (Moray 1967), or does the processor draw on the resources as needed (and as made available)? In the third case, the cognitive architecture would drive energy consumption and allocation, but the locus of control for the division of resources remains unspecified in any case. Presumably an "executive" function that either coordinates information processors drawing on different pools of resources or decides how resources will be allocated must itself consume resources, in terms of both energy required for decision making and mental "space" or structure required. Hence, resource theory does not solve the homunculus problem for theories of attention nor does it adequately describe the resource allocation strategies behind the performance of information-processing tasks.

4.5.1.3 Empirical Tests of the Model

Navon and Gopher (1979) commented on the problem of empirically distinguishing declines in performance due to insufficient supply from those resulting from increases in

* This is a curious historical development, since these relatively separate areas of research converge on the same psychological processes.

† The argument here is not that neural structures are not constrained by the laws of thermodynamics—clearly they are—but that thermodynamic principles implied by the metaphor are not sufficient for the development of a complete description of resources and their relation to cognitive activity.

demand. They asked, "When the performance of a task deteriorates, is it because the task now gets fewer resources or because it now requires more?" (Navon and Gopher 1979, p. 243). Navon and Gopher (1979) characterized the problem as distinguishing between changes in resources and changes in the subject-task parameters that constrain resource utilization. They offered two approaches to avoid this conundrum: One approach is to define the fixed constraints of a task and then observe how the information-processing system manages the processes within those constraints. The degree of freedom of the system, in this view, is the pool of resources available, in which the term resource is interpreted broadly to include quality of information, number of extracted features, or visual resolution. The subject-task parameters define what is imposed on the system (the demands) and the resources refer to what the system does in response to the demands (allocation of processing units). From this perspective, resources can be manipulated by the information-processing system within the constraints set by the subject-task parameters. A second approach is to distinguish the kind of control the system exerts on resources between control on the use of processing devices (what we have called structure) and control of the properties of inputs that go into these devices. The devices are *processing resources*. The other kind of control is exerted on *input resources*, which represents the flexibility a person has for determining which inputs are operated on, as determined by subject-task parameters. Processing resources are limited by the capacities of information processors, whereas input resources are limited by subject-task parameters (and allocation strategies that determine which information the operator attends to). Presumably the individual has some control over the allocation strategy, in terms of the processing resources devoted to a task, although these can also be driven by task demands (e.g., a spatial task requires spatial processing units). Navon and Gopher did not advocate either approach but presented both approaches as alternatives for further investigation. The implication for examining the resource model of stress is that one must manipulate both the subject-task parameters (e.g., by varying the psychophysical properties of the stimulus, manipulating the state of the observer, or varying the kind of information processing demanded by the task) and the allocation strategies used by the operator (the input resources, e.g., payoff matrices, task instructions). This would provide information regarding how specific stressors impair specific information-processing units and how they change a user's resource allocation strategies in the presence of stress that is continuously imposed on operators of complex computer-based systems.

In a later article, Navon (1984) moved to a position that is less favorable toward resources than his earlier approach, asserting that predictions derived by resource theory could be made, and results explained, without appealing to the resource concept at all (see also Rugg 1986). One could instead interpret effects in terms of the outputs of information processors. Most manipulations, such as task difficulty (which in Navon's view influences the efficiency of a unit of resources) or complexity (which affects the load, or the number of operations required), influence the demand for processing, with supply having no impact on their interaction. However, this approach assumes a clear distinction between outputs of a processing system and the concept of a resource, and Navon's (1984) notion of specific processors seems blurred with the notion of a resource, as both are utilized for task performance. Nevertheless, his critique regarding the vagueness of the resource concept is relevant, and Navon did argue that if resources are viewed as an intervening variable rather than a hypothetical construct the concept has utility.

4.5.1.4 Structural Mechanisms

If different kinds of information processing draw on different kinds of resources, in terms of the information processors engaged in a task, stressors may have characteristic effects on each resource. In addition, as Navon and Gopher (1979) have noted, an aspect of resource utilization is the efficiency of each resource unit. It may be that stress degrades the efficiency of information-processing units, independent of energy level or allocation strategy (cf. Eysenck and Calvo 1992). Investigation of such effects could be accomplished by transitioning between tasks requiring different kinds of information processing and determining if the effects of stress on one structure impacts the efficiency of a second structure.

The quality of resources can vary in terms of not only the kind of information-processing unit engaged but also the kind of task required. Following Rasmussen's (1983) classification system for behavior as a heuristic for design some tasks require knowledge-based processing, in which the operator must consciously rely on his or her mental model of the system in order to achieve successful performance. Other tasks fall under the category of rule-based behavior, in which a set of rules or procedures defines successful task response. The third category is skill-based behavior, in which a task is performed with a high degree of automaticity. Presumably each kind of task requires different amounts of resources, but they may also represent qualitatively different forms of resource utilization. In other words, these tasks may differ in the efficiency of a unit of resources as well as in effort allocation strategies. As task performance moves from knowledge- to rule- to skill-based processing (e.g., with training), the cognitive architecture may change such that fewer information-processing units are required and those that are engaged in the performance become more efficient. Moreover, the way in which each of these systems degrade with time under stress may be systematic with the more fragile knowledge-based processing degrading first, followed by rule-based processing, and skill-based processing degrading last (at this point, one may begin to see breakdown of not only psychological processes but also physiological ones; see Hancock and Warm 1989). This degradation may follow a hysteresis function such that a precipitous decline in performance occurs as the operator's resource capacity is reduced below a minimum threshold for performance. Moreover, these processes may recover in an inverse form with skill-based processing recovering first, followed by rule-based and knowledge-based processing.

Note that it may be difficult to distinguish "pure" knowledge-based processing from rule- or skill-based activity. An alternative formulation is the distinction between controlled and automatic processing (Schneider and Shiffrin 1977). Although originally conceived as categories, it is likely that individuals engaged in real-world tasks utilize both automatic and controlled processing for different aspects of performance and that for a given task there are levels of automaticity possible. Treating skills as a continuum rather than as discrete categories may be a more theoretically useful framework for quantifying resources and information processing, and thereby elucidating the effects of stress on performance.

4.5.1.5 Energetic Mechanisms

To investigate the energetic aspects of resources, one must manipulate environment-based perturbations, in the form of external stressors (noise, heat) and task demands, to systematically affect inflow versus outflow of energy. Presumably inflow is controlled by arousal levels, physiological energy reserves, and effort. One could examine performance under manipulations of energetic resources under dual-task performance (e.g., what happens to the performance on two tasks under conditions of sleep deprivation or caffeine consumption?). For example, the steady state can be perturbed by increasing (e.g., caffeine) or decreasing (e.g., sleep deprivation) energy while systematically varying the demands for the two tasks.

4.5.1.6 Structure and Energy

Another empirical challenge is to distinguish resources as structure from resources as energy. Given the definitional problems associated with the resource concept, it is not clear whether performance declines because of reduction in energy level or degradation of structures (i.e., failures or declines in the efficiency of processing units) or a combination of both. If structure and energy are distinct elements of resources it is hypothetically possible to manipulate one while holding the other constant, although the validity of this assumption is questionable. Is it possible to manipulate specific forms of information processing under constant energy level? Is it possible to manipulate energy level independent of which cognitive processes are utilized? If the decline in available resources is, at least in part, due to the degradation of particular information-processing units, then transferring to a task requiring the same processor should lead to worse performance than transferring to one that is different (cf. Wickens 1980, 1984). For instance, if a person engages in a task requiring verbal working memory while under stress and then transitions to a task requiring spatial discrimination, performance on the latter should depend only on energetic factors and not on structural ones. Note, however, that in this case the effects of different mental capacities would be confounded with the effects of novelty and motivation on performance.

4.5.1.7 Application of Neuroergonomics

The burgeoning field of neuroergonomics seeks to identify the neural bases of psychological processes involved in real-world human–technology interaction (Parasuraman 2003). As we state in Section 4.5 (Hancock and Szalma 2007), recent advances in neuroergonomics promise to identify cognitive processes and their link to neurological processes. For instance, the cognitive process of emotion regulation has been linked to genetic variation in the regulation of neuronal processes. Neuroscientists have isolated a particular genetic polymorphism, 5-HTTLPR, which moderates the level of an individual's emotional reactivity. The extent of emotional reaction influences the type and amount of mental resources mobilized for its regulation, which potentially signifies far-reaching effects for the entire HCI (Pezawas et al. 2005). Neuroergonomic research may therefore permit a more robust and quantitative definition of resources, although we caution that a simple reductionist approach is not likely to be fruitful as might initially be conceived (see Hancock and Szalma 2003b). In addition, the stress concept itself rests in part on more precise definitions of resources (Hancock and Szalma 2007). Thus, resolution of the resource issue with respect to cognitive processing and task performance would also clarify the workload and stress concepts. We therefore view neuroergonomics as one promising avenue for future research to refine the workload/stress and resource concepts.

4.5.2 Development of Adaptation under the Stress Model

Effective adaptation has always been key when performing in demanding environments. This section addresses how adaptation comes about.

4.5.2.1 Quantify the Task Dimensions

A major challenge for the Hancock and Warm (1989) model is the quantification of the base axes representing task dimensions. Specification of these dimensions is necessary if the vector representation postulated by Hancock and Warm is to be developed and if the resource construct is to be more precisely defined and quantified. However, task taxonomies that are general across domains present a theoretical challenge, because they require an understanding and quantification of how individuals process information along the spatial and temporal task dimensions, and how these change under stressful conditions. Quantification of information processing, and subsequent quantification of the base axes in the Hancock and Warm (1989) model, will permit the formalization of the vector representation of adaptive state under stress (see Figure 4.4).

4.5.2.2 Attentional Narrowing

Recall that Hancock and Weaver (2002) argued that the distortions of spatial and temporal perception have a common attentional mechanism. Two implications of this assertion are as follows: (1) Events (internal or external) that distort one dimension will distort the other and (2) these distortions are unlikely to be orthogonal. With very few exceptions, little research has addressed the possibility of an interaction between distortions of spatial and temporal perceptions in stressful situations on operator performance. Preliminary evidence suggests that these two dimensions may in fact not share a simple, common mechanism (Ross et al. 2003; Thropp, Szalma, and Hancock 2004), although further research is needed to confirm this finding. An additional important issue for empirical research is whether we are dealing with "time-in-memory" or "time-in-passing" (and to some extent, space-in-memory vs. space-in-passing). Thus, the way in which perceptions of space and time interact to influence operator state will depend on how temporal perceptions (and spatial perception, for that matter) are measured.

A possible explanation for perceptual distortions under conditions of heavy workload and stress concerns the failure to switch tasks when appropriate. Switching failures may be responsible for the observation in secondary task methodology that some participants have difficulty dividing their time between tasks as instructed (e.g., 70% to the primary task and 30% to the secondary task). This difficulty may result from the participant's inability to accurately judge how long he or she has attended to each task during a given time period. The degree to which distortions in the perception of space–time are related to impairments in task switching under stressful conditions and the degree to which these distortions are related to attention allocation strategies in a secondary task paradigm are questions for empirical resolution.

4.5.2.3 Stressor Characteristics

Even if space and time do possess a simple, common mechanism, it may be that specific stressors do not affect spatial and temporal perceptions in the same way. For instance, heat and noise may distort perception of both space and time but not to the same degree or in the same fashion. It is important to note that spatial and temporal distortions may *themselves* be appraised as stressful, as they might interfere with the information-processing requirements of a task. Consequently, some kinds of information processing might be more vulnerable to one or the other kind of perceptual distortion. Clearly, performance on tasks requiring spatial abilities, such as mental rotation, could suffer as a result of spatial distortion, whereas they might be unaffected (or, in some cases, facilitated) by temporal distortion. Other tasks, such as tasks that rely heavily on working memory or mathematical ability, or tasks requiring target detection, could each show different patterns of change in response to space–time distortion.

4.5.2.4 Potential Benefits of Space–Time Distortion

Under certain conditions, the narrowing of spatial attention can benefit performance through the elimination of irrelevant cues. The precise conditions under which this occurs, however, remains unclear. In addition, it is important to identify the circumstances under which time distortion might actually prove beneficial. Here, operators perceive that they have *additional* time to complete the task at hand (Hancock and Weaver 2005). This has great benefit in task performance situations where attentional narrowing is less likely to have deleterious effects. At this point in time, this is an empirical question that might be amenable to controlled testing.

4.5.2.5 Changes in Adaptation: The Roles of Time and Intensity

The degree to which a task or the physical and social environment imposes stress is moderated by the characteristics of the stimuli as well as the context in which events occur. However, two factors that seem to ubiquitously influence how much stress impairs adaptation are (appraised) intensity of the stressor and duration of exposure. We have reported meta-analytic evidence that these two factors jointly impact task performance across different orders of tasks (e.g., vigilance tasks, problem solving, tracking; see Hancock, Ross, and Szalma 2007; Szalma and Hancock 2011; Szalma, Hancock, and Quinn 2008). Duration is further implicated in information processing itself and may be a central organizing principle for information processing in the brain. Duration and intensity of environmental stimulation can likewise influence emotional reactions and consequently which emotion regulation or stress-coping strategy an individual opts to employ (Gross 1998, 2002). Empirical research is, however, still needed to explore programmatically the interactive effects of these variables across multiple forms of information processing.

4.5.3 Understanding Performance–Workload Associations/Dissociations

Factors that prompt associations or dissociations are herein discussed as well as their contribution to perceived workload.

4.5.3.1 Task Factors

Although Hancock (1996) and Yeh and Wickens (1988) have articulated the patterns of performance–workload relations and how these are diagnostic with respect to processing requirements, little systematic effort has been spent on further investigating these associations/dissociations. The primary question is what factors drive dissociations and insensitivities when they occur. For instance, for vigilance tasks mostly associations are observed, whereas for other tasks, such as those with high working memory demand, dissociations are more common (Yeh and Wickens 1988). Enhanced understanding of these relations would inform the Hancock and Warm (1989) model by permitting specification of the conditions under which individuals pass over the thresholds of failure at each level of person–environment transaction/adaptation.

4.5.3.2 Multidimensionality of Workload

To date, consideration of performance–workload dissociations has been primarily concerned with global measures of perceived workload. However, there is clear evidence that perceived workload is in fact multidimensional. For instance, vigilance tasks are characterized by high levels of mental demand and frustration (Warm, Dember, and Hancock 1996).

It is likely that the pattern of performance–workload links is different not only for different orders of performance (different tasks) but also for different dimensions of workload. One approach to addressing this question would be to systematically manipulate combinations of these two variables. For instance, if we consider performance in terms of detection sensitivity, memory accuracy, and speed of response, and the dimensions of workload defined by the National Aeronautics and Space Administration (NASA) Task Load Index (Hart and Staveland 1988), we could discuss how variations in memory load or discrimination difficulty link to each subscale.

4.6 INDIVIDUAL DIFFERENCES IN PERFORMANCE, WORKLOAD, AND STRESS

In previous work, we have reviewed the relations between individual differences in state and trait and efforts to quantify human information processing (Szalma and Hancock 2005). In this section, we address how individual differences (state and trait) are related to stress and coping.

4.6.1 Trait Differences

Individual differences research has been a neglected area in human factors and experimental psychology. Much of the early work on individual differences was done by researchers who were unconcerned with human–technology interactions to the extent that a bifurcation between two kinds of psychology occurred (Cronbach 1957). There is evidence, however, that affective traits influence information processing and performance. Thus, extraversion is associated not only with superior performance in working memory tasks and divided attention but also with poorer sustained attention (however, see Koelega 1992). Trait anxiety is associated with poor performance, although results vary across task types and contexts (Matthews, Deary, and Whiteman 2003; Szalma 2008). A possible next step for such research is to systematically vary task elements, as discussed previously (in Section 4.3.1) in the context of the Hancock and Warm model, and test hypotheses regarding how trait anxiety relates to specific task components. The theoretical challenge for such an undertaking is that it requires a good taxonomic scheme for tasks as well as a well-articulated theory of traits and performance. However, trait theories have neglected specific task performance, focusing instead on global measures (e.g., see Barrick, Mount, and Judge 2001), and there is a lack of a comprehensive theory that accounts for trait–performance relations (Matthews, Deary, and Whiteman 2003). Most current theories are more like frameworks that do not provide specific mechanisms for how personality impacts cognition and performance (e.g., see McCrae and Costa 1999). Although Eysenck (1967) proposed a theory of personality based on arousal and activation, which has found some support (Matthews and Gilliland 1999), there is also evidence to the end that arousal and task difficulty fail to interact as predicted (Matthews 1992). Eysenck's (1967) theory was also weakened by the general problems associated with arousal theory accounts for stress effects (Hockey 1984). An alternative formulation is that of Gray (1991) who argued for two systems, (1) one responding to reward signals and (2) one to punishment. The behavioral activation system (BAS) is associated with positive affect, whereas the behavioral inhibition system (BIS) is associated with negative affect. In a review and some comparisons of the Eysenck and Gray theories, Matthews and Gilliland (1999) partially supported both the theories but concluded that Gray's BAS/BIS distinction provides a superior match to positive and negative affect relative to Eysenck's arousal dimensions. Further, the BAS/BIS distinction accords with theories of approach/avoidance motivation (e.g., Elliot and Covington 2001). Indeed, intraindividual approaches to investigating the complex interplay between stress, coping, and performance outcomes are hailed as the most promising methodology (Folkman et al. 1986). There are also theories that focus on a particular trait such as extraversion (Humphreys and Revelle 1984) or trait anxiety (Eysenck and Calvo 1992). Although useful, such specific theories do not encompass other traits or interactions among traits. Such interactive effects can influence cognitive performance and perceived stress and workload (Szalma et al. 2005). These interactions should be further studied with an eye to linking them to information-processing theories.

4.6.2 Affective State Differences

It is intuitive that stress would induce more negative affective states and that traits would influence performance via an effect on states. For instance, one would expect that trait anxiety would influence performance because high trait anxious individuals experience state anxiety more frequently than those low on that trait. Although such mediation effects are observed, there is also evidence that for certain processes, such as hypervigilance to threat, trait anxiety is a better predictor of performance than state anxiety (Eysenck 1992). In terms of appraisal theory, traits may influence the form and content of appraisals, as well as the coping skills an individual can deploy to deal with stress. With respect to adaptation, it is likely that individual differences in both trait and state will influence adaptation, both behavioral and physiological, by affecting the "width" of the plateau of effective adaptation at a given level and by changing the slope of decline in adaptation when the adaptation threshold is reached. That is, higher skill levels protect from declines in adaptive function by increasing the threshold for failure at a given level (i.e., comfort, performance, physiological response). The modification of the Hancock and Warm (1989) model illustrating these individual differences in effects is shown in Figure 4.6 (and see Szalma 2008). Multiple frameworks of state dimensions exist, but most focus on either two (e.g., Thayer 1989; Watson and Tellegen 1985) or three (Matthews et al. 1999, 2002) frameworks. In the context of task performance, Matthews and his colleagues have identified three broad state dimensions reflecting the cognitive, affective, and motivational aspects of an individual's current psychological state.

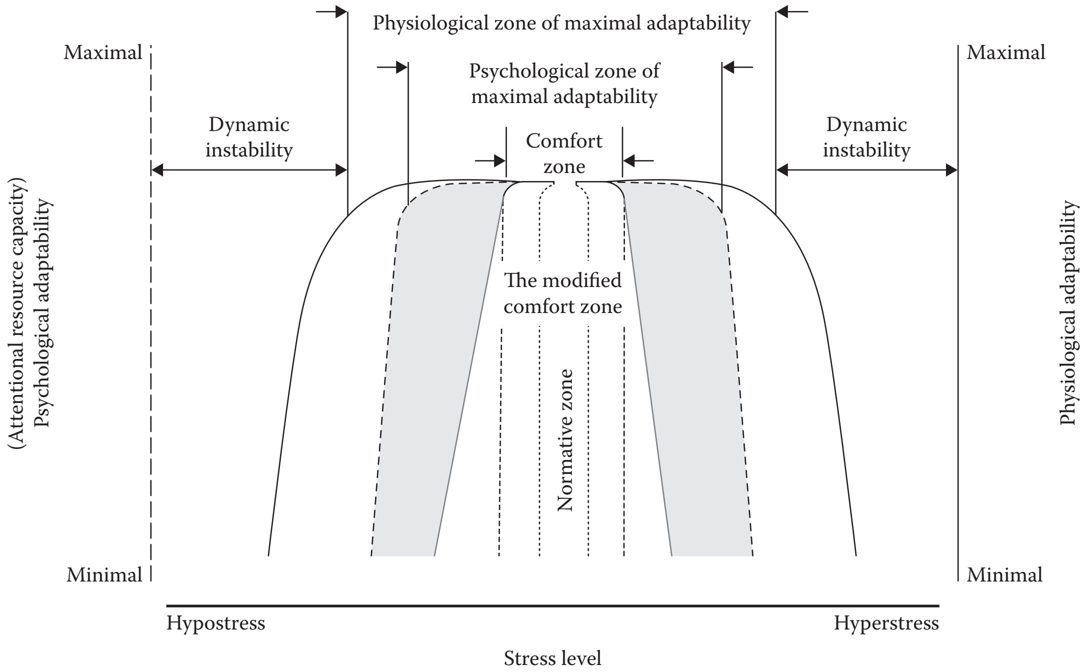

Physiological zone of maximal adaptability
Maximal
Maximal
Psychological zone of maximal adaptability
Dynamic instability
Dynamic instability
Comfort zone
(Attentional resource capacity)
Psychological adaptability
The modified comfort zone
Physiological adaptability
Normative zone
Minimal
Minimal
Hypostress
Hyperstress
Stress level

FIGURE 4.6 Modification of the adaptability model shown in Figure 4.3. The adaptability model of Hancock and Warm (1989) shown in Figure 4.3 has been modified to illustrate how individual differences may influence stress and adaptation. It is likely that cognitive and affective traits influence the width of both the comfort and performance zones (i.e., the thresholds for declines in adaptation) as well as the rate of decline in adaptability when a threshold is crossed. For instance, individuals high in trait anxiety likely have a narrower plateau of stability and therefore manifest lower thresholds for discomfort and performance degradation than individuals low on that trait. Further, the rate of decline in adaptation may increase as a function of trait anxiety.

These dimensions are "worry," which reflects the cognitive dimension of stress, and "task engagement" and "distress," which reflect the affective, cognitive, and motivational components of state. Specifically, a high level of distress is indicative of overload in processing capacity, and task engagement reflects a theme of commitment to effort (Matthews et al. 2002). Matthews and his colleagues have demonstrated that changes in task demand influence the pattern of stress state. Should affective state fail to detrimentally influence task performance itself, it can still critically impact consequential levels of "psychophysiological activation, strain, and fatigue aftereffects" (Robert and Hockey 1997, p. 73). It is therefore important to incorporate assessment of operator state into the interface design process so that the interaction with technology fosters task engagement and minimizes distress and worry.

4.6.2.1 Attentional Narrowing and Adaptive Response

As with other aspects of perception, there are individual differences in the perception of space and time (Hancock and Weaver 2005; Wachtel 1967). Further, because the subjective experience of stress is often multidimensional, it may be that although two individuals are subjectively stressed by the same situation, their stress profiles differ. Affective states can likewise influence the extent of attentional allocation. Affective states high in motivational intensity, either pleasant or unpleasant, lead to a narrowing of attentional focus, whereas affective states low in motivational intensity, again regardless of valence, cause attentional broadening (Gable and Harmon-Jones 2009). Individuals are also likely to differ in the strategies they employ to cope with the distortions of space–time and emotional flux they experience while in a stressful environment, and these coping differences, if they exist, might depend on the quality (e.g., noise, heat, low signal salience) and source (e.g., the environment, the task) of the stress and the personality traits of an individual.

4.6.2.2 Hedonomics and Individual Differences

In addition to application of individual differences research to the development of training or selection procedures, individual difference variables can be used to promote hedonomic approaches to design and facilitate interface design. Thus, if the traits that influence the subjective experience of an interaction with technology are identified, that interface can then be configured to meet the preferences and the trait/state profile of an individual user and promote positive affective states. However, for such efforts to succeed, the relations among traits and cognitive, perceptual, and motor performance need to be established via theory-guided empirical research.

4.7 IMPLICATIONS OF STRESS FOR RESEARCHERS AND PRACTITIONERS

For both research and design applications, extant research on stress and performance indicates that assessments of workload and affective state are important for a more complete understanding of HCI. Such assessments can aid in identifying which components of an interface or task are appraised as stressful, and thus interfaces can be designed to mitigate their negative effects. For instance, research is needed to establish which task parameters control the patterns of performance–workload associations and dissociations and how these change dynamically as a function of time on task. The Hancock and Warm (1989) model of stress established general task dimensions (space–time) that influence stress state and behavioral adaptability, but the metrics for these dimensions remain elusive. This problem results from the central issue regarding how to quantify human information processing (Hancock, Szalma, and Oron-Gilad 2005) and define "mental resources" more precisely (Hancock and Szalma 2007). Efforts to resolve these definitional problems would improve stress theory and its application to interface design. Future research should therefore examine the relations between task dimensions and user characteristics and how these change over time and under high-stress conditions.

In addition to changing the task, there are other techniques that can be applied to the design of human–computer interfaces for use in stressful environments. These include skill development (e.g., Hancock 1986), use of configural displays (Hancock and Szalma 2003a; Wickens 1996), as well as use of technologies employing adaptive automation and decision aids (Hancock and Chignell 1987). With respect to skill development in particular, an area in need of research is how individuals who develop expertise in a task also learn how to cope with stress while performing the task. In order to understand how individuals accomplish this, one is required to understand in depth how different forms of stress influence different forms of information processing. Intuitively, automation and decision aids seem to be key tools for relieving stress during task performance. Although experiments have yielded some promising results (Funke et al. 2007), further research is necessary to determine this supposition under different kinds of stress as such technologies could merely serve to divert attentional resources away from the task.

It is also important for both researchers and practitioners to consider the characteristics of a user and how these characteristics interact with the task or interface to influence performance.

An understanding of how individual differences influence HCI can facilitate the development of tailored training regimens as well as interfaces that can more effectively adapt to the user. Systems that can respond to changes in operator affective state can achieve the desired human–machine synergy in HCI (cf. Hancock 2009). Realizing these goals, however, will require adequate theory development and subsequent empirical research to determine the nature of the relations among the person and environmental variables. It is particularly important to design interfaces that permit autonomous motivation (Deci and Ryan 2000) and to understand how operators of computer-based systems can internalize extrinsic motivation as they gain experience on the task (Gagne and Deci 2005). We suggest here that researchers and designers identify the motivational affordances in the task environment and utilize them to enhance the experience of HCI and improve overall system performance under stress. Motivational affordances will be elements of the work environment that facilitate and nurture intrinsic motivation. Particularly important for design will be the identification of motivational invariants, which are those environmental factors that consistently determine an individual's level of intrinsic (or extrinsic) motivation across contexts. Careful analyses of the motivational affordance structure will permit design of tasks that are more likely to be enjoyable by rendering the tools convivial (Illich 1973) and thereby facilitating human–machine synergy (see Hancock 1997).

4.8 SUMMARY AND CONCLUSIONS

In this chapter, we review theories of stress and performance and their relevance to human–technology interaction. We also show that despite being developed in separate research traditions, workload and stress can be viewed as different perspectives on the same fundamental problem. We outline general principles for stress mitigation and discuss issues that require further research. Of particular importance are establishing sound measures of information processing and mental resource expenditure as well as articulating the relevant task dimensions and how they trigger self-regulatory mechanisms, specifically emotion (stress) regulation techniques. Given that stress can be understood only in relation to the transaction between an individual and the environment, it is crucial to establish how trait and state characteristics of the individual influence their appraisals. Finally, it is important in practical applications to treat stress at multiple levels, ranging from the physiological to the organizational sources of adverse performance effects. Different emotion regulation strategies attempt to mitigate stress at these various levels; which techniques are chosen by an operator to utilize can significantly influence the success of an HCI. Traditional attempts to treat stress problems unidimensionally will continue to fail until the person, task, and the physical and social/organizational environment are treated by analysis as a coherent system. Researchers and practitioners in HCI should therefore expand their efforts beyond the design of displays and controls of interfaces and include assessments of person-related factors that influence performance as well as the design of the physical and social environment in which an HCI occurs.

REFERENCES

Ackerman, P. L., and E. D. Heggestad. 1997. Intelligence, personality, and interests: Evidence for overlapping traits. *Psychol Bull* 121:219–45.

Aspinwall, L. G., L. Richter, and R. R. Hoffman. 2002. Understanding how optimism works: An examination of optimists' adaptive moderation of belief and behavior. In *Optimism and Pessimism: Implications for Theory, Research, and Practice*, ed. E. C. Chang, 217–38. Washington, DC: American Psychological Association.

Bandura, A. 1997. *Self-Efficacy: The Exercise of Control*. New York: W.H. Freeman and Company.

Barrick, M. R., M. K. Mount, and T. A. Judge. 2001. Personality and performance at the beginning of the new millennium: What do we know and where do we go next? *Pers Perform* 9:9–29.

Bennett, K. B., and J. M. Flach. 1992. Graphical displays: Implications for divided attention, focused attention, and problem solving. *Hum Factors* 34:513–52.

Bless, H. 2001. Mood and the use of general knowledge structures. In *Theories of Mood and Cognition: A User's Guidebook*, ed. L. L. Martin and G. L. Clore, 9–26. Mahwah, NJ: Erlbaum.

Brown, J. M., and E. A. Campbell. 1994. *Stress and Policing: Sources and Strategies*. Chichester, UK: Wiley.

Bursill, A. E. 1958. The restriction of peripheral vision during exposure to hot and humid conditions. *Q J Exp Psychol* 10:113–29.

Carver, C. S., and M. F. Scheier. 1998. *On the Self-Regulation of Behavior*. New York: Cambridge University Press.

Conway, G. E., J. L. Szalma, and P. A. Hancock. 2007. A quantitative meta-analytic examination of whole-body vibration effects on human performance. *Ergonomics* 50:228–45.

Cornsweet, D. M. 1969. Use of cues in the visual periphery under conditions of arousal. *J Exp Psychol* 80:14–8.

Cronbach, L. J. 1957. The two disciplines of scientific psychology. *Am Psychol* 12:671–84.

Csikszentmihalyi, M. 1990. *Flow: The Psychology of Optimal Experience*. New York: Harper & Row.

Deci, E. L., and R. M. Ryan. 1985. *Intrinsic Motivation and Self-Determination in Human Behavior*. New York: Plenum Press.

Deci, E. L., and R. M. Ryan. 2000. The "what" and "why" of goal pursuits: Human needs and the self-determination of behavior. *Psychol Inq* 11:227–68.

Deci, E. L., R. M. Ryan, and R. Koestner. 1999. A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychol Bull* 125:627–68.

Dirkin, G. R., and P. A. Hancock. 1984. Attentional narrowing to the visual periphery under temporal and acoustic stress. *Aviat Space Environ Med* 55:457.

Dirkin, G. R., and P. A. Hancock. 1985. An attentional view of narrowing: The effect of noise and signal bias on discrimination in the peripheral visual field. In *Ergonomics International 85: Proceedings of the Ninth Congress of the International Ergonomics Association*, ed. I. D. Brown, R. Goldsmith, K. Coombes, and M. A. Sinclair, 751–3. Bournemouth, England, London: Taylor & Francis.

Driskell, J. E., J. H. Johnston, and E. Salas. 2001. Does stress training generalize to novel settings? *Hum Factors* 43:99–110.

Easterbrook, J. A. 1959. The effect of emotion on cue utilization and the organization of behavior. *Psychol Rev* 66:183–201.

Elliot, A. J., and M. V. Covington. 2001. Approach and avoidance motivation. *Educ Psychol Rev* 13:73–92.

Ericsson, K. A. 2006. The influence of experience and deliberate practice on the development of superior expert performance. In *The Cambridge Handbook of Expertise and Expert Performance*, ed. K. A. Ericsson, N. Charness, R. R. Hoffman, and P. J. Feltovich, 683–703. Cambridge: Cambridge University Press.

Eysenck, H. J. 1967. *The Biological Basis of Personality*. Springfield, IL: Charles C. Thomas.

Eysenck, M. W. 1992. *Anxiety: The Cognitive Perspective*. Hillsdale, NJ: Erlbaum.

Eysenck, M. W., and M. G. Calvo. 1992. Anxiety and performance: The processing efficiency theory. *Cogn Emot* 6:409–34.

Flach, J., P. A. Hancock, J. K. Caird, and K. Vicente, eds. 1995. *Global Perspectives on the Ecology of Human-Machine systems*, vol. 1. Hillsdale, NJ: Erlbaum.

Folkman, S., R. S. Lazarus, C. Dunkel-Schetter, A. DeLongis, and R. J. Gruen. 1986. Dynamics of a stressful encounter: Cognitive appraisal, coping, and encounter outcomes. *J Pers Soc Psychol* 50:992–1003.

Fromm, E. 1976. *To Have or to be?* New York: Harper & Row.

Funke, G., G. Matthews, J. S. Warm, and A. K. Emo. 2007. Vehicle automation: A remedy for driver stress? *Ergonomics* 50:1302–23.

Gable, P., and E. Harmon-Jones. 2009. The blues broaden, but the nasty narrows: Attentional consequences of negative affects low and high in motivational intensity. *Psychol Sci* 20:1–5.

Gagne, M., and E. L. Deci. 2005. Self-determination theory and work motivation. *J Organ Behav* 26:331–62.

Gibson, J. J. 1966. *The Senses Considered as Perceptual Systems*. Boston: Houghton Mifflin.

Gibson, J. J. 1979. *The Ecological Approach to Visual Perception*. Boston: Houghton Mifflin.

Gray, J. A. 1991. Neural systems, emotion, and personality. In *Neurobiology of Learning, Emotion, and Affect*, ed. J. Madden IV, 273–306. New York: Raven Press.

Gross, J. J. 1998. Antecedent- and response-focused emotion regulation: Divergent consequences for experience, expression, and physiology. *J Pers Soc Psychol* 74:224–37.

Gross, J. J. 2002. Emotion regulation: Affective, cognitive, and social consequences. *Psychophysiology* 39:281–91.

Gross, J. J., and R. W. Levenson. 1993. Emotional suppression: Physiology, self-report, and expressive behavior. *J Pers Soc Psychol* 64:970–86.

Gross, J. J., J. M. Richards, and O. P. John. 2006. Emotion regulation in everyday life. In *Emotion Regulation in Couples and Families: Pathways to Dysfunction and Health*, ed. D. K. Snyder, J. A. Simpson, and J. N. Hughes, 13–35. Washington, DC: American Psychological Association.

Haga, S. M., P. Kraft, and E. K. Corby. 2009. Emotion regulation: Antecedents and well-being outcomes of cognitive reappraisal and expressive suppression in cross-cultural samples. *J Happiness Stud* 10:271–91.

Hancock, P. A. 1986. The effect of skill on performance under an environmental stressor. *Aviat Space Environ Med* 57:59–64.

Hancock, P. A. 1996. Effects of control order, augmented feedback, input device and practice on tracking performance and perceived workload. *Ergonomics* 39:1146–62.

Hancock, P. A. 1997. *Essays on the Future of Human-Machine Systems*. Eden Prairie, MN: BANTA Information Services Group.

Hancock, P. A. 2009. *Mind, machine and morality*. Chichester, UK: Ashgate.

Hancock, P. A. 2010. The battle for time in the brain. In *Time, Limits and Constraints: The Study of Time XIII*, ed. J. A. Parker, P. A. Harris, and C. Steineck. Leiden, the Netherlands: Brill.

Hancock, G. M., and G. F. Beatty. 2010. The effects of automatic and deliberate emotion regulation on sustained motor force production. *Program of the University of Florida's 2010 Graduate Student Council Interdisciplinary Research Conference*, 39 (Abstract Only). Gainesville, FL: UF's Graduate Student Council.

Hancock, P. A., and J. K. Caird. 1993. Experimental evaluation of a model of mental workload. *Hum Factors* 35:413–29.

Hancock, P. A., and M. H. Chignell. 1987. Adaptive control in human-machine systems. In *Human Factors Psychology*, ed. P. A. Hancock, 305–45. Amsterdam: Elsevier.

Hancock, P. A., and G. R. Dirkin. 1983. Stressor induced attentional narrowing: Implications for design and operation of person-machine systems. *Proc Hum Factors Assoc Can* 16:19–21.

Hancock, G. M., and P. A. Hancock. 2010. Can technology create instant experts? *Ergonomist* 460:4–5.

Hancock, P. A., A. A. Pepe, and L. L. Murphy. 2005. Hedonomics: The power of positive and pleasurable ergonomics. *Ergon Des* 13:8–14.

Hancock, P. A., J. M. Ross, T. Oron-Gilad, and J. L. Szalma. 2005. *The Incorporation of Comprehensive Thermal Stress Effects into IMPRINT* (Tech. Rep. No. 1). Orlando, FL: University of Central Florida, Minds in Technology, Machines in Thought Laboratory.

Hancock, P. A., J. M. Ross, and J. L. Szalma. 2007. A meta-analysis of performance response under thermal stressors. *Hum Factors* 49:851–77.

Hancock, P. A., and J. L. Szalma. 2003a. Operator stress and display design. *Ergon Des* 11:13–8.

Hancock, P. A., and J. L. Szalma. 2003b The future of neuroergonomics. *Theor Issues Ergon Sci* 4:238–49.

Hancock, P. A., and J. L. Szalma. 2007. Stress and neuroergonomics. In *Neuroergonomics: The Brain at Work*, ed. R. Parasuraman and M. Rizzo, 195–206. Oxford: Oxford University Press.

Hancock, P. A., J. L. Szalma, and T. Oron-Gilad. 2005. Time, emotion, an the limits to human information processing. In *Quantifying Human Information Processing*, ed. D. K. McBride and D. Schmorrow, 157–75. Lanham, MD: Lexington Books.

Hancock, P. A., and J. S. Warm. 1989. A dynamic model of stress and sustained attention. *Hum Factors* 31:519–37.

Hancock, P. A., and J. L. Weaver. 2005. On time distortion under stress. *Theor Issues Ergon Sci* 6:193–211.

Harris, W. C., P. A. Hancock, and S. C. Harris. 2005. Information processing changes following extended stress. *Mil Psychol* 17:115–28.

Hart, S. G., and L. E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Human Mental Workload*, ed. P. A. Hancock and N. Meshkati, 139–83. Amsterdam: Elsevier.

Hassenzahl, M., and D. Ullrich. 2007. To do or not to do: Differences in user experience and retrospective judgments depending on the presence or absence of instrumental goals. *Interact Comput* 19:429–37.

Hockey, R. 1984. Varieties of attentional state: The effects of environment. In *Varieties of Attention*, ed. R. Parasuraman and D. R. Davies, 449–83. New York: Academic Press.

Hockey, G. R. J. 1997. Compensatory control in the regulation of human performance under stress and high workload: A cognitive–energetical framework. *Biol Psychol* 45:73–93.

Hockey, G. R. J., A. W. K. Gaillard, and M. G. H. Coles, eds. 1986. *Energetic Aspects of Human Information Processing*. The Netherlands: Nijhoff.

Hockey, R., and P. Hamilton. 1983. The cognitive patterning of stress states. In *Stress and Fatigue in Human Performance*, ed. G. R. J. Hockey, 331–62. Chichester: Wiley.

Humphreys, M. S., and W. Revelle. 1984. Personality, motivation, and performance: A theory of the relationship between individual differences and information processing. *Psychol Rev* 91:153–84.

Illich, I. 1973. *Tools for Conviviality*. New York: Harper & Row.

Johnston, J. H., and J. A. Cannon-Bowers. 1996. Training for stress exposure. In *Stress and Human Performance*, ed. J. E. Driskell and E. Salas, 223–56. Mahwah, NJ: Erlbaum.

Kahneman, D. 1973. *Attention and Effort*. Englewood Cliffs, NJ: Prentice Hall.

Kasser, T. 2002. *The High Price of Materialism*. Cambridge, MA: MIT Press.

Koelega, H. S. 1992. Extraversion and vigilance performance: Thirty years of inconsistencies. *Psychol Bull* 112:239–58.

Lang, P. J. 1995. The emotion probe: Studies of motivation and attention. *Am Psychol* 50:372–85.

Lazarus, R. S. 1991. *Emotion and Adaptation*. Oxford: Oxford University Press.

Lazarus, R. S. 1999. *Stress and Emotion: A New Synthesis*. New York: Springer.

Lazarus, R. S., and S. Folkman. 1984. *Stress, Appraisal, and Coping*. New York: Springer Verlag.

Lepper, M. R., and D. I. Cordova. 1992. A desire to be taught: Instructional consequences of intrinsic motivation. *Motiv Emot* 16:187–208.

Luczak, H., M. Roettling, and L. Schmidt. 2003. Let's talk: Anthropomorphization as means to cope with stress of interacting with technical devices. *Ergonomics* 46:1361–74.

Matthews, G. 1992. Extraversion. In *Handbook of Human Performance, vol. 3: State and Trait*, ed. A. P. Smith and D. M. Jones, 95–126. London: Academic Press.

Matthews, G. 2001. Levels of transaction: A cognitive science framework for operator stress. In *Stress, Workload, and Fatigue*, ed. P. A. Hancock and P. A. Desmond, 5–33. Mahwah, NJ: Erlbaum.

Matthews, G., I. J. Deary, and M. C. Whiteman. 2003. *Personality Traits*. 2nd ed. Cambridge: Cambridge University Press.

Matthews, G., and K. Gilliland. 1999. The personality theories of H. J. Eysenck and J. A. Gray: A comparative review. *Pers Individ Dif* 26:583–626.

Matthews, G., L. Joyner, K. Gilliland, S. Campbell, S. Falconer, and J. Huggins. 1999. Validation of a comprehensive stress state questionnaire: Towards a state 'big three'? In *Personality Psychology in Europe*, ed. I. Mervielde, I. J. Deary, F. De Fruyt, and F. Ostendorf, vol. 7, 335–50. Tilburg, The Netherlands: Tilburg University Press.

Matthews, G., L. Joyner, K. Gilliland, S. Campbell, S. Falconer, J. Huggins, K. Gilliland, R. Grier, and J. S. Warm. 2002. Fundamental dimensions of subjective state in performance settings: Task engagement, distress, and worry. *Emotion* 2:315–40.

Mauss, I. B., S. A. Bunge, and J. J. Gross. 2007. Automatic emotion regulation. *Soc Personal Psychol Compass* 1:146–67.

McCrae, R. R., and P. T. Costa. 1986. Personality, coping, and coping effectiveness in an adult sample. *J Pers* 54:385–405.

McCrae, R. R., and P. T. Costa. 1999. A five-factor theory of personality. In *Handbook of Personality: Theory and Research*, 2nd ed., ed. L. A. Pervin and O. P. John, 139–53. New York: Guilford Press.

Moray, N. 1967. Where is capacity limited? A survey and a model. *Acta Psychol* 27:84–92.

Moray, N., ed. 1979. *Mental Workload: Its Theory and Measurement*. New York: Plenum Press.

Navon, D. 1984. Resources—A theoretical soupstone? *Psychol Rev* 91:216–34.

Navon, D., and D. Gopher. 1979. On the economy of the human information processing system. *Psychol Rev* 86:214–55.

Norman, D., and D. Bobrow. 1975. On data-limited and resource-limited processing. *J Cogn Psychol* 7:44–60.

Ochsner, K. N., and J. J. Gross. 2005. The cognitive control of emotion. *Trends Cogn Sci* 9:242–9.

O'Donnell, R. D., and F. T. Eggemeier. 1986. Workload assessment methodology. In *Handbook of Human Perception and Performance, vol. II: Cognitive Processes and Performance*, ed. K. R. Boff, L. Kaufman, and J. P. Thomas, 42-1–42-49. New York: Wiley.

Parasuraman, R. 2003. Neuroergonomics: Research and practice. *Theor Issues Ergon Sci* 4:5–20.

Parasuraman, R., and P. A. Hancock. 2001. Adaptive control of mental workload. In *Stress, Workload, and Fatigue*, ed. P. A. Hancock and P. A. Desmond, 305–20. Mahwah, NJ: Erlbaum.

Penley, J. A., and J. Tomaka. 2002. Associations among the big five, emotional responses, and coping with acute stress. *Pers Individ Dif* 32:1215–28.

Pezawas, L., A. Meyer-Lindenberg, E. M. Drabant, B. A. Verchinksi, K. E. Munoz, B. S. Kolachana, M. F. Egan, V. S. Mattay, A. R. Hariri, and D. M. Weinberger. 2005. 5-HTTLPR polymorphism impacts human cingulate-amygdala interactions: A genetic susceptibility mechanism for depression. *Nat Neurosci* 8:828–34.

Pilcher, J. J., E. Nadler, and C. Busch. 2002. Effects of hot and cold temperature exposure on performance: A meta-analytic review. *Ergonomics* 45:682–98.

Rasmussen, J. 1983. Skills, rules, and knowledge: Signals, signs, and symbols, and other distinctions in human performance models. *IEEE Trans Syst Man Cybern* SMC-13:257–66.

Reed, E. S. 1996. *Encountering the World: Toward an Ecological Psychology*. Oxford: Oxford University Press.

Robert, G., and G. R. J. Hockey. 1997. Compensatory control in the regulation of human performance under stress and high workload: A cognitive-energetical framework. *Biol Psychol* 45:73–93.

Ross, J. M., J. L. Szalma, J. E. Thropp, and P. A. Hancock. 2003. Performance, workload, and stress correlates of temporal and spatial task demands. *Proc Hum Factors Ergon Soc* 47:1712–6.

Rugg, M. D. 1986. Constraints on cognitive performance: Some problems with and alternatives to resource theory. In *Energetics and Human Information Processing*, ed. G. R. J. Hockey, A. W. K. Gaillard, and M. G. H. Coles, 353–71. Dordrecht: Martinus Nijhoff Publishers.

Ryan, R. R., and E. L. Deci. 2000. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *Am Psychol* 55:66–78.

Ryan, R. R., and E. L. Deci. 2001. On happiness and human potentials: A review of research on hedonic and eudaimonic well-being. *Annu Rev Psychol* 52:141–66.

Saunders, T., J. E. Driskell, J. Johnston, and E. Salas. 1996. The effect of stress inoculation training on anxiety and performance. *J Occup Health Psychol* 1:170–86.

Scerbo, M. W., F. G. Freeman, and P. J. Mikulka. 2003. A brain-based system for adaptive automation. *Theor Issues Ergon Sci* 4:200–19.

Scheier, M. F., and C. S. Carver. 1985. Optimism, coping, and health: Assessment and implications of generalized outcome expectancies. *Health Psychol* 4:219–47.

Scherer, K. R. 1999. Appraisal theory. In *Handbook of Cognition and Emotion*, ed. T. Dalgleish and M. Power, 638–63. New York: Wiley.

Schneider, W., and R. M. Shiffrin. 1977. Controlled and automatic human information processing I: Detection, search, and attention. *Psychol Rev* 84:1–66.

Seligman, M. E. P., and M. Csikszentmihalyi. 2000. Positive psychology: An introduction. *Am Psychol* 55:5–14.

Selye, H. 1976. *The Stress of life. (Revised edition)*. New York: McGraw-Hill.

Speisman, J. C., R. S. Lazarus, A. Mordkoff, and L. Davison. 1964. Experimental reduction of stress based on ego- defense theory. *J Abnorm Soc Psychol* 68:367–80.

Szalma, J. L. 2008. Individual differences in stress reaction. In *Performance Under Stress*, ed. P. A. Hancock and J. L. Szalma, 323–57. Aldershot, England: Ashgate.

Szalma, J. L., and P. A. Hancock. 2005. Individual differences in information processing. In *Quantifying Human Information Processing*, ed. D. K. McBride and D. Schmorrow, 177–93. Lanham, MD: Lexington Books.

Szalma, J. L., and P. A. Hancock. 2011. Noise effects on human performance: A meta-analytic synthesis. *Psychological Bulletin* 137:682–707.

Szalma, J. L., P. A. Hancock, and S. Quinn. 2008. A meta-analysis of the effect of time pressure of human performance. *Hum Factors Ergon Soc Annu Meet Proc* 52:1513–6.

Szalma, J. L., J. S. Warm, G. Matthews, W. N. Dember, E. M. Weiler, A. Meier, and F. T. Eggemeier. 2004. Effects of sensory modality and task duration on performance, workload, and stress in sustained attention. *Hum Factors* 46:219–33.

Thayer, R. E. 1989. *The Biopsychology of Mood and Arousal*. New York: Oxford University Press.

Thropp, J. E., J. L. Szalma, and P. A. Hancock. 2004. Performance operating characteristics for spatial and temporal discriminations: Common or separate capacities? *Proc Hum Factors Ergon Soc* 48:1880–4.

van Reekum, C., and K. R. Scherer. 1997. Levels of processing in emotion-antecedent appraisal. In *Cognitive Science Perspectives on Personality and Emotion*, ed. G. Matthews, 259–300. Amsterdam: Elsevier.

Venkatesh, V. 2000. Determinants of perceived ease of use: Integrating control, intrinsic motivation, and emotion into the technology acceptance model. *Inf Syst Res* 11:342–65.

Vicente, K. J., and J. Rasmussen. 1992. Ecological interface design: Theoretical foundations. *IEEE Trans Syst Man Cybern* 22:589–606.

Wachtel, P. L. 1967. Conceptions of broad and narrow attention. *Psychol Bull* 68:417–29.

Warm, J. S., W. N. Dember, and P. A. Hancock. 1996. Vigilance and workload in automated systems. In *Automation and Human Performance: Theory and Applications*, ed. R. Parasuraman and M. Mouloua, 183–200. Hillsdale, NJ: Erlbaum.

Warm, J. S., R. Parasuraman, and G. Matthews. 2008. Vigilance requires hard mental work and is stressful. *Hum Factors* 50:433–41.

Watson, D., and A. Tellegen. 1985. Toward a consensual structure of mood. *Psychological Bulletin* 98:219–235.

Wickens, C. D. 1980. The structure of attentional resources. In *Attention and Performance VIII*, ed. R. Nickerson, 239–57. Hillsdale, NJ: Erlbaum.

Wickens, C. D. 1984. Processing resources in attention. In *Varieties of Attention*, ed. R. Parasuraman and D. R. Davies, 63–102. New York: Academic Press.

Wickens, C. D. 1996. Designing for stress. In *Stress and Human Performance*, ed. J. E. Driskell and E. Salas, 279–95. Mahwah, NJ: Erlbaum.

Yeh, Y., and C. D. Wickens. 1988. Dissociation of performance and subjective measures of workload. *Hum Factors* 30:111–20.

Yerkes, R. 1918. Psychology in relation to the war. *Psychol Rev* 25:85–115.

Young, M. S., and N. A. Stanton. 2002. Malleable attentional resources theory: A new explanation for the effects of mental underload on performance. *Hum Factors* 44:365–75.

Zhang, T., D. B. Kaber, B. Zhu, M. Swangnetr, P. Mosaly, and L. Hodge. 2010. Service robot feature design effects on user perceptions and emotional responses. *Intell Serv Rob* 3:73–88.

Zimmerman, B. J. 2000. Attaining self-regulation: A social cognitive perspective. In *Handbook of Self-Regulation*, ed. M. Boekaerts, P. R. Pintrch, and M. Zeidner, 13–39. San Diego: Academic Press.

5 Choices and Decisions of Computer Users

Anthony Jameson

CONTENTS

5.1 INTRODUCTION

5.1.1 CONCEPTS AND GOALS

Computer users are constantly making small choices and larger decisions about how to use their computing technology, such as the following:

- Which of the available photo management apps shall I use on my smartphone?
- Shall I dictate this e-mail message using speech recognition or tap in the text with a stylus?
- How should I configure my privacy settings?

This chapter focuses on cases, like these, in which a user can choose among two or more *options*, none of which is correct or incorrect but one of which can be *preferred* to the others. The term *preferential choice* will be used to distinguish this situation from *nonpreferential* choices that concern the correct way to operate a system, such as "Which of these unfamiliar icons do I have to click on to send off my e-mail message?”

We will use the terms *choice* and *decision*, together and in alternation, to do justice to the variety of forms that the processes in question can take. *Decision* suggests a thorough, effortful process, whereas *choice* suggests a quick selection that may be based, for example, on habit. Both types of process occur in computer users, often with regard to the same set of options.

The following are the goals of this chapter:

1. Bring preferential choices and decisions of computer users into the foreground as a topic in human–computer interaction (HCI).
2. Provide access to the relevant psychological and HCI literature by summarizing key concepts and results and listing references.
3. Provide a framework for thinking about how to help computer users make better preferential choices and decisions.

5.1.2 Relationships to Other Human–Computer Interaction-Related Research

Figure 5.1 visualizes the relationships between these goals and the goals of three other broad types of research that fall within or overlap with the HCI field.

5.1.2.1 Interaction Design Guidelines and Principles; Help and Training

Much of what is known about how to design interactive systems and their associated help and training material can be seen as concerning ways of helping users to make the right choices: to click on the right icon or web link, select the correct command from a menu, or identify the part of the system that will provide the needed functionality. Interaction designers have become skilled at helping users to make these choices well, for example, by designing effective visual displays, making the user’s options clearly identifiable and understandable, providing informative feedback on the user’s actions, and making the actions reversible in case they do not yield a satisfactory result (see, e.g., Johnson 2010, for a collection of well-known sets of user-interface design guidelines). Similarly, those who develop online help and training programs have worked out a rich set of best practices for instructing and advising users about the choices that they need to make. Most of the content of help and training concerns the general question of how to operate the system in question, but some of it explicitly addresses preferential choices, such as when to use each of two available methods for accomplishing a particular goal or what type of configuration is best under what circumstances (e.g., “This setting is recommended if you often work off-line”).

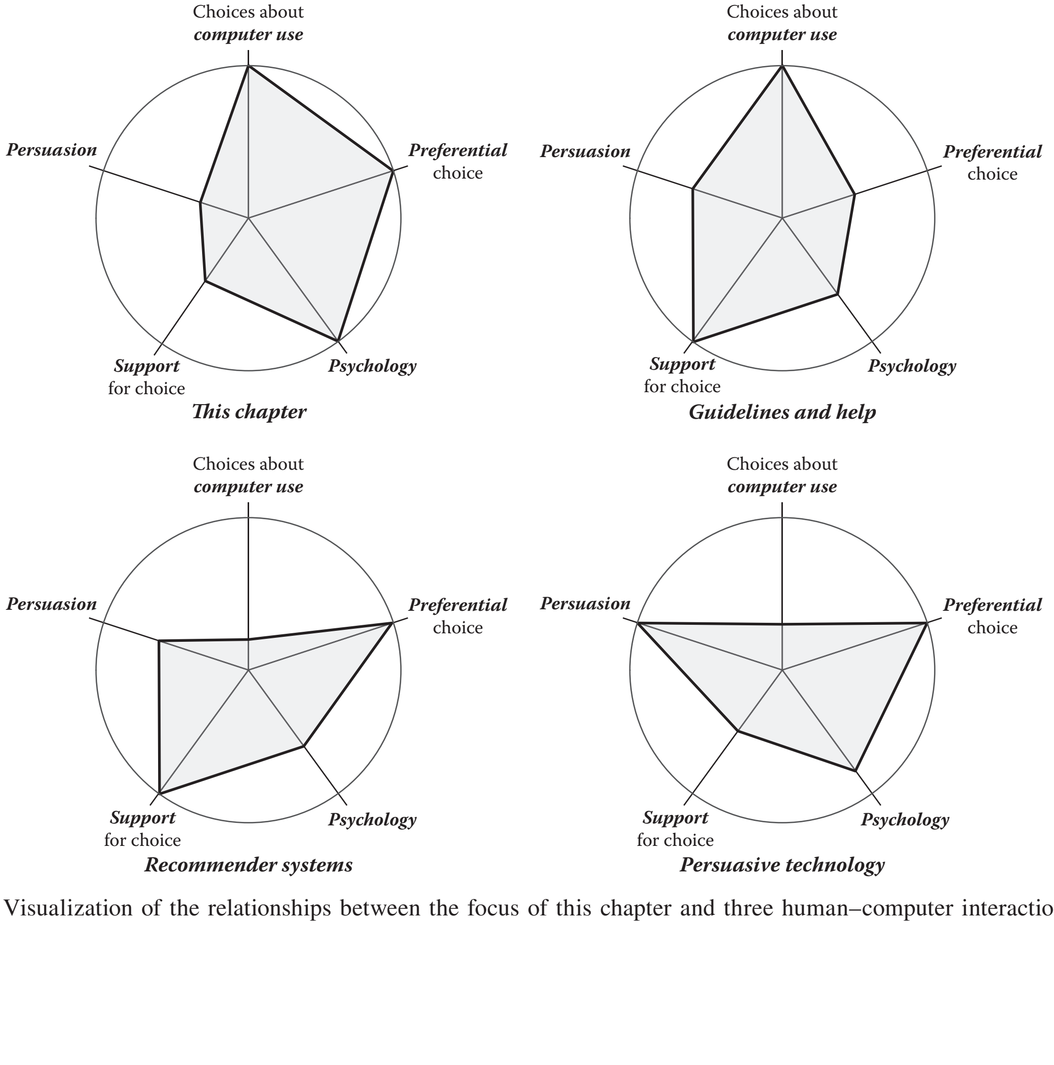


FIGURE 5.1 Visualization of the relationships between the focus of this chapter and three human–computer interaction-related areas of research.

Guidelines and design principles are rarely tailored explicitly to supporting preferential choices and decisions, and the related research hardly ever refers to the psychological literature on these topics that is covered in this chapter.

5.1.2.2 Recommender Systems

A focus on preferential choice and decisions is found, by contrast, in research on recommender systems (see, e.g., Jannach et al. 2011; Ricci et al. 2010) which aim to support and influence users' choices concerning products to buy, documents to read, and a variety of other types of items. As Figure 5.1 shows, recommender systems almost always support decisions that are not about the use of computing technology as such. The work in this area tends to be based to some extent on knowledge about psychological processes involved in preferential choice, but the main focus of attention is on accurately predicting what items will satisfy a user, rather than on understanding and influencing the user's decision-making processes.

5.1.2.3 Persuasive Technology

Yet another line of research (see, e.g., Fogg 2003; Fogg, Cueller, and Danielson 2008) differs from the previous paradigm mainly in its emphasis on motivating and persuading people to do some particular thing (e.g., save energy), which either that person or someone else has decided is best for the person in question. This line of research has yielded a wealth of ideas about how computing technology can be deployed to influence people's beliefs and behaviors. But only a few of the choices and behaviors targeted for persuasion (e.g., none of the 12 "domains for persuasive technology" listed in Table 7.1 of Fogg, Cueller, and Danielson 2008) concern computer use as such.

As Figure 5.1 indicates, this chapter will not go into much depth on the question of how to support and influence preferential choices concerning computer use. Instead, by foregrounding this class of choices and by providing an introduction to the large areas of relevant psychological literature, it aims to encourage and support increased attention to this topic.* Systematic efforts to support choices and decisions of this type should be able to benefit greatly from appropriately adapted knowledge transferred from the other three areas of research, notwithstanding the various differences visualized in Figure 5.1.

5.1.3 Preview of Aspects of Preferential Choice and Decision Making

Figure 5.1 reflects the fact that psychological research about how people make preferential choices and decisions has received limited attention in HCI so far.† One reason may be the fact that there is no single relevant theory in psychology that could be straightforwardly adapted to the needs of the HCI field. Although dozens of books and hundreds of articles from relevant psychological research exist, they come from several research traditions that only partly overlap and refer to each other. The discussion in this chapter will draw from these areas: judgment and decision making (see, e.g., Hastie and Dawes 2010; Koehler and Harvey 2004; Lichtenstein and Slovic 2006; Schneider and Shanteau 2003; Newell, Lagnado, and Shanks 2007; Weber and Johnson 2009); naturalistic decision making (Klein 1998), the reasoned action approach (Fishbein and Ajzen 2010), research on habitual behavior (Wood and Neal 2007), behavioral economics (Ariely 2008; Iyengar 2010; Thaler and Sunstein 2008), and research on self-control (Rachlin 2000) and on compliance tactics (Cialdini 2007).

As a way of providing a reasonably coherent overview despite the differences among these research traditions and their terminologies, Table 5.1 lists the aspects of choice and decision processes that will be covered in turn in this chapter, formulating each one in terms of one or more "questions" that a computer user might conceivably "ask" him- or herself while considering a choice or decision. Although in some cases such questions may be consciously asked and addressed by a computer user, the processing represented in the table by a question often occurs without any verbal formulation or conscious deliberation—whatever particular definition of the elusive concept of *consciousness* one may prefer to use (see, e.g., Wilson 2002).

With any given choice or decision for a particular person, in general only some subset of these considerations will be relevant, and the table is not intended to convey a particular temporal order of processing: Because of the variety of forms that

TABLE 5.1
Preview of the Aspects of Preferential Choice and Decision Making Discussed in This Chapter

Topic	Questions That a Decision Maker May Consider
Focusing on goals and values	What is a good decision-making process for this situation? What are my relevant goals and values?
Situation assessment and option identification	What's going on in this situation? What are my options?
Anticipation of consequences	What would the consequences be if I chose this option? How desirable would they be?
Intertemporal choice	How should I value consequences that will not occur until sometime in the future? How should I deal with a sequence of repetitions of basically the same choice?
Reuse of previous choices	What did I choose the last time I had a choice like this?
Social influence	What do other people choose in this situation? What do they want or expect me to choose?
Learning from experience	What can I learn from the results of the choice that I have made?

* A first step toward a systematic approach to supporting preferential choice on the basis of the conceptual framework of this chapter is offered by Jameson et al. (2011).

† Two thorough book-length syntheses of cognitive psychology research for HCI (Gardiner and Christie 1987; Johnson 2010) include hardly any references to the sort of psychology literature cited in this chapter.

preferential choices and decisions can take, it would not be realistic to try to formulate a causal model or a process model, for example, in the form of a flowchart, though models of this sort are often found useful for particular types of choice or decision-making situation (see, e.g., Wickens and Hollands 2000, Chapter 7; Fishbein and Ajzen 2010; Klein 1998, Chapter 3).

5.2 GENERAL PREFERENTIAL CHOICE PROBLEMS

Although opportunities to make preferential choices and decisions crop up constantly with just about every type of interactive system, there are three generic classes of choice that are worth distinguishing, because of their frequency of occurrence and because they have attracted a fair amount of attention in HCI research. Table 5.2 introduces them to facilitate reference to them at various points later in the chapter.

5.2.1 Decision about Whether to Use a Given System

One type of decision that a person can make with regard to computer use is that of whether to use a given system at all. The most extensive line of research that has looked into this question is research on *technology acceptance*. A good entry point to this literature is the influential article by Venkatesh et al. (2003), which presented the Unified Theory of Acceptance and Use of Technology (UTAUT), a model that integrates eight previously developed models, including the especially widely studied *technology acceptance model* (see, e.g., Venkatesh and Davis 2000). These models in turn drew their inspiration from more general theories from social psychology and sociology, such as the precursors of the recently formulated *reasoned action approach* of Fishbein and Ajzen (2010).

TABLE 5.2
Three General Types of Preferential Choice That Have Been Studied in Human–Computer Interaction

Generic Choice Problem	Selected Research Issues
Decision about whether to use a given system	What variables influence people's decisions about whether to use a given system if it is made available to them (usually: within an organization)? What are the causal relationships among these variables? How can these variables be measured?
Choice of a method from a set of alternative methods	When more than one method is available for a particular subtask, how do users decide which one to use? Why do even experienced users sometimes persist in using inefficient methods?
Configuration decision	How do people decide whether and when to configure an application? What difficulties do they encounter when making configuration choices?

Table 5.3 gives an impression of the basic nature of the models in this area by depicting the four main variables in the UTAUT model that influence intention to use a given system and actual use of the system, along with examples of questionnaire items typical of those used to measure these variables. The model also includes claims about several variables that moderate the influence of these main variables: *gender, age, experience*, and *voluntariness of use*.

Although some of these questions are reminiscent of questions from usability scales such as System Usability Scale (SUS) (Brooke 1996), the overall goal of the model and the associated measuring instruments are not to assess usability but rather to predict whether potential users (typically, employees in a given company) will actually use a given system (e.g., a new videoconferencing system) if it is made available to them. Note that most of the questions related to the variables *social influence* and *facilitating conditions* concern considerations other than usability.

Researchers and practitioners in the HCI field usually want to go beyond *predicting* whether people in a given target group will use a given (type of) system, to attempt to improve the system (and/or related resources) to increase the likelihood that the system will be used and the success of its use. Still, the large amount of information collected in the technology acceptance area about variables related to choices about system use and about ways of measuring these variables can help to stimulate and structure thinking about this class of choices. Researchers in this area regularly introduce new variables and new perspectives that shed light on different aspects of acceptance decisions (see, e.g., Bagozzi 2007; Loraas and Diaz 2009).

TABLE 5.3
The Four Main Variables in the UTAUT Model and Typical Questionnaire Items Used to Measure Them

Performance expectancy
- Using the system in my job would enable me to accomplish tasks more quickly.
- Using the system would improve my job performance.
- Using the system would make it easier to do my job.

Effort expectancy
- Learning to operate the system would be easy for me.
- My interaction with the system would be clear and understandable.
- I would find the system to be flexible to interact with.

Social influence
- People who influence my behavior think that I should use the system.
- People who are important to me think that I should use the system.

Facilitating conditions
- I have control over using the system.
- I have the resources necessary to use the system.
- I have the knowledge necessary to use the system.
- The system is not compatible with other systems I use.

Source: Based on parts of Figure 3 and Tables 9–12 of Venkatesh, V., M. G. Morris, G. B. Davis, and F. D. Davis. 2003. *MIS Quart* 27(346): 425–78.

5.2.2 Choice of a Method

In all but the simplest interactive systems, there is often more than one method available for achieving a given goal. Whenever the user can choose freely between two or more methods, the choice is preferential. Card, Moran, and Newell (1983) introduced in their Goals, Operators, Methods, and Selection Rules (GOMS) model (described most completely in Card, Moran, and Newell 1983; see also Kieras 2008) a notation for such cases: the two or more available methods are described as part of the model for a given task, and it is assumed that each user has learned a selection rule for making the choice (e.g., "Use the mouse instead of the cursor keys if the target is more than a couple of inches away on the screen"); this assumption is plausible given that the GOMS model assumes that users have considerable experience with the system and the tasks in question.

In the intervening years, some research has looked at the ways in which users learn selection rules on the basis of experience with the methods in question (see, e.g., Gray and Boehm-Davis 2000) and at the considerations that users take into account when choosing among methods (see, e.g., Young and MacLean 1988; Jameson and Klöckner 2005), whereas other researchers have investigated situations in which users systematically fail to use suitable methods that are available to them (Carroll and Rosson 1987; Bhavnani and John 2000; Bhavnani, Peck, and Reif 2008; Charman and Howes 2003).

5.2.3 Configuration Decision

A usually more complicated type of choice that users can make concerns whether, when, and how to configure an application to suit their own tastes and needs. Over the years, researchers have repeatedly found this type of problem to be challenging for most users (see, e.g., Mackay 1991; McGrenere, Baecker, and Booth 2007), and it has attracted increased attention in recent years because of the practically important problem of configuring privacy settings in social network platforms (see, e.g., Iachello and Hong 2007).

5.3 FOCUSING ON GOALS AND VALUES

The first of the general considerations listed in Table 5.1 concerns the basic values that a chooser will be guided by when making a choice. Although computer users often do not think explicitly about these values, interaction designers ought to be aware of them when considering how to support good choices; and calling these issues to the user's attention may be an effective tactic.

5.3.1 What Constitutes a Good Choice or Decision?

The most fundamental question is that of what constitutes a good choice in the first place. Before considering what choosers think about this issue, we should notice a shift in the thinking of scientists who have studied decision making. Traditional notions of what constitutes a good decision are that a decider should (1) apply a decision procedure that is normatively justifiable (e.g., consistent with the laws and principles of logic, probability, and expected utility) and (2) choose the action that will maximize desirable (and minimize undesirable) outcomes under idealized conditions (see, e.g., Gigerenzer and Todd 1999, Chapter 1; Gigerenzer 2007, Chapter 5). More recently, researchers have become impressed by the extent to which animals and humans can function quite effectively by using decision procedures that are justifiable only in the sense that they work well in the environment in which they are applied and make good use of the decider's limited time and cognitive resources. For example, a web searcher's strategy of clicking on the first link on the search result page that looks reasonably relevant would be hard to justify in terms of a normatively optimal general strategy; but if the user's previous experience with the search engine in question has shown that the first reasonably relevant-looking link is almost always the best one, this strategy can be considered *ecologically rational* for that search engine. The same point can apply to the decision rule of always buying your smartphone applications from your favorite vendor or always accepting the default configuration when installing new software. In cases where the choices of a computer user make sense only given particular assumptions about the structure of the environment, the best way to help the user make good choices may be to ensure that the environment fulfills these assumptions.

Researchers have also investigated the question of what constitutes a good decision process from the point of view of the decision maker (see, e.g., Bettman, Luce, and Payne 2006; Hastie 2001; Yates, Veinott, and Patalano 2003). Although specific answers to this question vary, the following statements are widely accepted:

1. Choosers want their decision to yield a good outcome.

   This point is not as straightforward as it may seem, because what counts as a good outcome depends in turn on a variety of factors, as we will see.
2. Choosers do not want to invest time and effort in the decision-making process itself that is out of proportion to the benefits of doing so.

   For example, when installing a new application, a user who is asked which specific components should be installed may choose the option "Everything" simply to save the time of deciding about the individual components, since the possible benefits of choosing any other option (e.g., saving a few megabytes of hard disk space) do not seem to justify the investment of even a few seconds of decision time.
3. Choosers prefer to avoid unpleasant thoughts.

   Some ways of thinking about a decision can involve distressing thoughts, as when a driver faces a choice between (1) ignoring an incoming text message from his boss and (2) driving less safely for a while in order to respond to the message. A user may be motivated to think about the decision in a way that avoids such thoughts (e.g., by convincing himself that he can respond to the boss's message without taking the slightest risk).

4. Choosers often want to be able to justify the decision that they have made to other persons—or to themselves.

   Justifiability is often simply a necessary condition for being able to implement a decision (cf. Lerner and Tetlock 2003). For example, even if a business person would really like to buy an iPhone for professional use, they may prefer a Blackberry instead because they think that this choice is more likely to be approved by their company's purchasing department. But even just the desire to convince another person or oneself that a decision was sound can cause people to look for justifiable decisions (see, e.g., Shafir, Simonson, and Tversky 2006).

   Consequently, one way of supporting preferential choice is to make it easy for the user to come up with a satisfying justification of whatever option is best for him, for example, by supplying a justification explicitly (as is done by many recommender systems) (see Tintarev and Masthoff 2010) or by structuring the situation in such a way that a justification is easy to derive.

5.3.2 Current Goals and Values

One characteristic of preferential choice is its dependence on the particular goals that the chooser is currently focusing on (see, e.g., Schneider and Barnes 2003). To a certain extent, this dependence is obviously necessary and appropriate: Your choice of an application to prepare a text document with should depend on whether you want it to be beautifully formatted or whether you just want to get it finished as quickly as possible. But the dependence on current goals can also lead to some curious phenomena: Both anecdotal evidence and some research (e.g., Iachello and Hong 2007, Section 3.3.2; Mackay 1991) concerning configuration decisions tell us that users often accept the default configuration of a system until some negative event (e.g., a privacy violation or a need to repeat a given tedious operation multiple times) prompts them to change the configuration. A normatively more rational way of deciding when and what to configure would involve something like estimating the total (discounted) benefit of the improved configuration over an extended period of system use. In contrast, reactive configuration can be seen as a response to the goal of preventing the specific negative thing that just happened from ever happening again. Whether this configuration action is really a good idea in the long run will depend on how well the short-term goal happens to coincide with the user's larger pattern of goals and use situations. Mackay (1991) and Iachello and Hong (2007) offer perceptive discussions of strategies for dealing with this type of discrepancy.

Keeney (1992) discusses in great depth the importance of ensuring that decisions depend on the decision maker's true values rather than on temporarily salient considerations such as those that are suggested by the set of options that are immediately available. Although interaction designers rarely, if ever, have an opportunity to support their users with in-depth decision analysis, calling the user's attention to important goals and values on a much smaller scale does represent a promising way of supporting preferential choice. Two experiments by Mandel and Johnson (2006) demonstrate clearly how a goal or value (e.g., "safety" or "economy" for a prospective car buyer) can be activated by a change in interface design (e.g., the colored background of the web pages of an e-commerce site), mostly without awareness on the part of the user.

5.4 SITUATION ASSESSMENT AND OPTION IDENTIFICATION

To be able to make a choice or decision, the chooser must normally in some sense be aware of the fact that a choice is available—though in extreme cases the awareness can be minimal, as when the choice is made out of habit or when it involves accepting the status quo or default option by doing nothing.

In experimental laboratory studies, the way in which the chooser perceives or "frames" the choice problem is largely under the control of the experimenter. Some well-known and striking results concern the effects on choice of the way in which the problem is framed. For example, people tend to be influenced strongly by whether options are described in terms of people being "saved" versus people "dying," even when the situations described in these terms are objectively identical. An important part of one of the dominant theories of judgment and decision making, *prospect theory* (originally presented by Kahneman and Tversky 1979), concerns the process of "editing" the initial representation of a choice problem to arrive at the chooser's own representation; but choosers often stick with the initial representation.

Like laboratory experimenters, interaction designers often have control over the way in which a choice is presented to the user. For example, users who purchase a software product are often offered an option like "Check this box to receive news about updates and special offers," which a user may mentally edit into a representation like "Check this box to get even more spam."

When decision making occurs outside the laboratory, the presentation of the choice problem is often less clear-cut; understanding the situation and identifying the available options can be a complex process (often called *situation assessment*) that calls for considerable expertise. This process has been extensively studied within the research paradigm of *naturalistic decision making* (see, e.g., Klein 1998; Klein 2008; Maule 2010). This type of decision making is typified by the situation of a fire brigade arriving at the scene of a burning building: The problem situation is changing rapidly over time, even as the decision makers contemplate how to deal with the fire; there is considerable stress because of the high stakes and because of environmental factors such as noise and heat; and on the positive side, the decision makers typically possess considerable experience in dealing with such situations, which makes it unnecessary for them to analyze the problem from first principles. Some key results of this research will be summarized below in Section 5.7.1. For now, the main point is that recognizing the need for a choice and identifying or generating one or more options is sometimes the most important and challenging aspect of a decision problem.

An implication for interaction design is that we should look out for situations in which recognizing and interpreting a decision situation may be unnecessarily or unduly challenging for at least some users. For example, a sophisticated user who installs a new web browser is likely to recognize the need to choose security and privacy settings that are well adapted to the context in which the browser will be used; a less sophisticated user is likely to accept the default settings, perhaps without even being aware that a choice exists.

In fact, the widespread tendency of people to overlook or ignore choice opportunities and accept the default represents a major way in which *choice architects* (to use the suggestive term of Thaler and Sunstein 2008), including interaction designers, can influence choices. Widely discussed controversies concerning computer use include the bundling of software with the Windows operating system (which offers new users a convenient default option for many application choices that they would otherwise have to make) and the default privacy options for social network platforms like Facebook. Outside of the arena of computer use, one of the primary and most successful tactics of interventions based on behavioral economics (such as the *libertarian paternalism* of Thaler and Sunstein 2008) is to provide a default option which is thought to be in the best interest of the people making the choice in question or of society as a whole (e.g., laws that state that every person can be viewed as an organ donor unless they have specified otherwise; see Johnson and Goldstein 2006).

5.5 ANTICIPATION OF CONSEQUENCES

The most dominant traditional view of decision making is a *consequentialist* one: that of a person who contemplates the (perhaps uncertain) consequences of choosing each of the available options and bases the decision on an evaluation of those consequences. As Table 5.1 indicates, there are other considerations that can affect a decision, and in fact, choosers sometimes do not contemplate consequences at all.

Still, computer users do sometimes anticipate the possible consequences of their choices, and one question that arises is that of what sorts of consequence they anticipate. If computer users were concerned only about traditional usability criteria, they might make their decisions solely on the basis of consideration of consequences like those covered by UTAUT's *performance expectancy* and *effort expectancy* variables (Table 5.3). The growing interest in recent years in a broader view of user experience (see, e.g., Law et al. 2009; Kuniavsky 2010) can be viewed as an awareness of a wider range of types of consequence that can influence users' evaluations of systems and possible actions.

5.5.1 Anticipating Experience

But how accurately can computer users anticipate the consequences of options? Even just anticipating the enjoyableness of an experience that has been described to you (e.g., using an allegedly delightful photo management app on a smartphone) is not as straightforward as it would intuitively seem. Trying the experience out briefly (e.g., with a demo version of the app) is not always a reliable test, partly because of people's tendency to adapt their tastes and expectations on the basis of new experience (see, e.g., Wilson 2002, Chapter 7). And if a user's initial expectation is (erroneously) that an experience will not be positive, he or she may refrain from trying it out in the first place.

A straightforward effort of designers to support the anticipation of the experience of performing an action is found in promises such as "Filling in our customer satisfaction questionnaire will take just 2 minutes of your time" or "Configuring the application is quick and easy." But this method presupposes that the user is likely to believe claims like these. An alternative approach is to consider nonverbal ways of previewing the consequences of an action. This general strategy has been explored extensively in the area of persuasive technology (see, e.g., Fogg 2003, Chapter 4), as with the "Baby Think it Over" infant simulator, which helps teen-aged girls anticipate realistically what it is like to take care of a baby. Some further work will probably be required before this strategy can be applied widely to (1) decisions concerning computer use and (2) decisions where it is not a priori clear which option is best for the chooser—that is, where the chooser must really *choose*, as opposed to being persuaded (cf. Figure 5.1).

5.5.2 Anticipating the Consequences of Configuration Choices

One challenge for users in connection with the configuration of applications (Mackay 1991; Iachello and Hong 2007) is that the consequences of configuration actions tend to be hard to anticipate. First, there is the question of how time-consuming, tedious, and risky the configuration actions themselves will be. Then there is the fact that the consequences of a configuration decision are often not immediately visible; they consist in changes to the computing environment that will have consequences in the future which will in turn depend on actions of the user and other configuration settings.

Gabrielli and Jameson (2009), applying an adapted heuristic walkthrough to parts of four widely used applications, found that about three-quarters of the formulations used to describe configuration options (e.g., "Accept cookies from third parties") did not appear to convey to a typical user a clear idea of the meaning of an option, the consequences of choosing it, or the overall desirability of choosing it. The proportion of problematic cases diminished to about one-half if the help texts explaining the options were taken into account.

5.6 INTERTEMPORAL CHOICE

5.6.1 Time Discounting

Humans and animals alike tend to prefer a benefit that will come soon to an equal benefit that will occur later in time. That is, they *discount* future benefits. For example, a member of an online community may be more willing to make a contribution if it appears on a web page immediately, so that its positive consequences (which can take various forms) occur without delay. As is the case with monetary investments, there are various good reasons to discount temporally distant benefits (including uncertainty about whether they will actually come about). A straightforward design implication is that, to encourage a user to choose a particular option, you can try to arrange for its benefits to come sooner rather than later. This strategy was applied by McDowell et al. (2003) to encourage nontechnical users to annotate HTML data for semantic web services.

But there are some more subtle aspects of time discounting that deserve attention.* These can be illustrated with reference to the frequent situation in which a person can choose between (1) an option that will bring a small benefit soon and (2) another option that will yield a larger benefit at a later time. If people's discount curves were exponential—as is the case with typical discount rates for financial investments and in early normative models of time discounting (see, e.g., Read 2004)—then people would always show *time consistency* in their preference between the smaller-sooner and the larger-later option: If, when asked on Monday, you prefer a larger benefit on Saturday afternoon to a smaller benefit on Friday afternoon, then you will express the same preference on Friday morning. As is illustrated in Figure 5.2, the discounting curves in question never cross.

Many studies with humans and animals have shown, however, that discounting curves are better described by a hyperbolic function (Figure 5.3) than by an exponential one. One implication of the mathematical form of a hyperbolic function (see, e.g., Read 2004) is that the curves in a problem like the one we are considering can cross. Concretely, in our example, when Friday morning arrives and the small benefit could be obtained almost immediately, the chooser may change his mind and opt for the smaller benefit after all. This particular type of *preference reversal* has been documented countless times in studies with animals (e.g., pigeons) and humans (see, e.g., Rachlin 2000, Chapter 2), and it corresponds to our everyday experience that benefits which are tangibly near can loom disproportionately large.

Often, people are aware of the danger of such a last-minute preference reversal and are willing to avoid it by *committing* themselves at an early point in time to the option with the larger-later benefit (Rachlin 2000, Chapter 3). One strategy is to eliminate the option with the smaller-sooner benefit (e.g., by permanently discontinuing membership in an online community that offers immediate but trivial rewards). A softer commitment mechanism involves arranging for a punishment or other disadvantage to be associated with the smaller-sooner option (e.g., throwing away your password for the online community in question, though you know you can always get a new one with some effort). A drawback of the softer mechanisms is that people may still succumb to the temptation of the smaller-sooner benefit and willingly accept the associated punishment, in which case they are worse off than they would have been without the commitment.

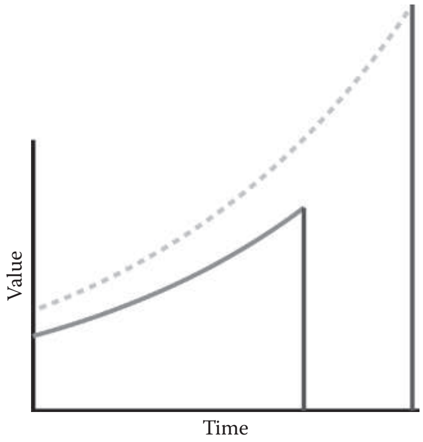


FIGURE 5.2 Exponential time discount functions for a smaller-sooner and a larger-later benefit. (Each of the vertical line segments on the right represents the value of a benefit at the point in time at which it occurs. Each curve represents the discounted value of the anticipated benefit at an earlier point in time.)

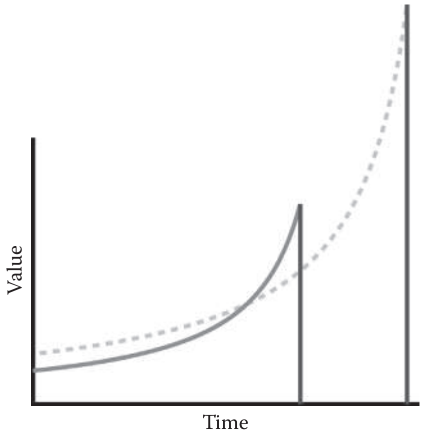


FIGURE 5.3 Hyperbolic time discount functions that cross. (Compare with Figure 5.2: The larger-later benefit is preferred until shortly before the time at which the smaller-sooner benefit will occur.)

One very general strategy for helping users to make better choices is to make available suitable commitment mechanisms. Many of the strategies applied within the paradigm of persuasive technology can be seen as ways of helping people to stick to a commitment that they have made (e.g., to exercise regularly). Where the choices in question concern computer use (which is normally not the case in the persuasive technology paradigm), there are additional forms

* Useful collections of articles on phenomena that arise when choices and/or their consequences are distributed over time have been edited by Loewenstein and Elster (1992) and by Loewenstein, Read, and Baumeister (2003).

of commitment mechanisms available, because the decision environment is more under the control of the designer and the user. For example, mechanisms that are commonly used to make it impossible for children to visit certain websites or to use certain applications can also be used as self-control mechanisms that people can willingly apply to themselves.

5.6.2 Choice Bracketing

The choice between a smaller-sooner and a larger-later benefit is actually quite straightforward compared with many situations that arise when options and their consequences are distributed over time. One key concept is that of *choice bracketing* (Read, Loewenstein, and Rabin 2006). Although the concept is actually more general, we will discuss only *temporal bracketing*, which is illustrated graphically in Figure 5.4.

The issue arises when a chooser confronts a sequence of similar choices—for example, which of two alternative keyboards to use to enter text on a smartphone: the traditional QWERTY keyboard or an unfamiliar keyboard that has been optimized for one-handed text input. Conceivably, a user could make this choice separately every time it arises, which would be an example of *narrow bracketing*. If instead the user opts for *broad bracketing*, she will think in terms of a general policy, such as the choice between: (1) "Always use the QWERTY keyboard"; (2) "Always use the alternative keyboard"; or (3) "Use the alternative keyboard when you have a lot of text to enter." Research has brought to light a number of typical advantages of broad bracketing, most of which are illustrated by this example.

One benefit is that a sequence of choices can have important properties that the chooser cannot see when contemplating the individual choices. For example, if the user consistently uses the alternative keyboard, she will initially enter text more slowly and with greater mental effort than with the QWERTY keyboard; but if she persists long enough, the alternative keyboard will eventually become easier and faster to use than the QWERTY keyboard. Similarly, the user's tastes can change: she will probably find the appearance of the alternative keyboard less strange and distracting.

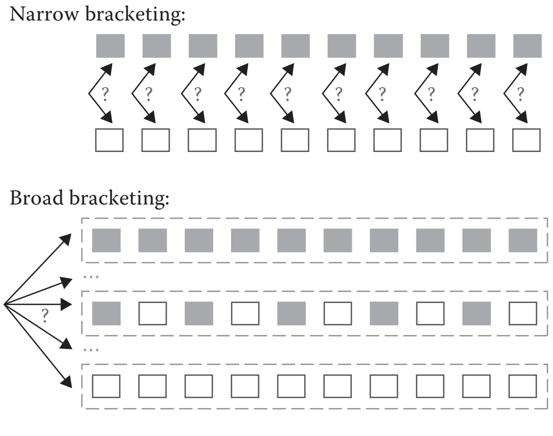


FIGURE 5.4 Visualization of the distinction between narrow and broad *temporal choice bracketing.*

Another emergent property of a sequence of choices is the amount of variety associated with it: A user might prefer to alternate between the use of a trackball and the use of a mouse to avoid one-sided use of her hand and arm muscles.

Situations where broad bracketing is possible may also involve time discounting: The user in our example might opt for narrow bracketing because she heavily discounts the long-term benefits of using the alternative keyboard. But the issues just discussed cannot all be reduced to time discounting. Rachlin (2000) uses the terms *complex ambivalence* and *simple ambivalence*, respectively, to distinguish the two cases.

Designers of interactive systems have many opportunities to encourage broad bracketing in cases where doing so seems conducive to good decision making. For example, instead of making two different virtual keyboards readily available at all times, the designer can make the choice of keyboard a configuration option—perhaps one that is difficult to change—so as to encourage the user to take a broader view. Conveying a realistic idea of the consequences associated with broadly bracketed options is more of a challenge, because by definition these consequences cannot be experienced immediately. In particular, the general strategy of trying something out to see if you like it is relatively hard to apply in cases where broad bracketing is appropriate.

5.7 REUSE OF PREVIOUS CHOICES

Sections 5.5 and 5.6 have shown that making choices on the basis of anticipated consequences can be an effortful and error-prone process. These considerations help to explain why choosers often apply a simpler general strategy: Choose the same option that you chose the last time you were in this situation, maybe adapting it a bit. Several complementary lines of psychological research help to understand how and why choices are often repeated.

5.7.1 Recognition-Primed Decision Making

The concept of *recognition-primed decision making* was developed by Klein and collaborators in the context of their studies of naturalistic decision making (introduced in Section 5.4).

On the basis of previous research on decision making, most of it in the laboratory, Klein expected that decision makers such as fireground commanders would typically consider two possible courses of action before deciding which one to execute. They were surprised to find that usually, the "decision makers" seemed not to be making decisions at all: Most often, they would evaluate the situation confronting them, remember a course of action that they had previously applied in one or more similar situations, and proceed to implement that action. As Table 5.4 indicates, a somewhat more complex variant of this basic procedure was observed in cases where a contemplated action was not obviously appropriate: The decision maker would anticipate the consequences of the action by a process called *mental simulation* and if necessary modify it until the mental simulation produced a satisfactory result. In

TABLE 5.4
Three Forms of Recognition-Primed Decision Making

Straightforwardly retrieve an action
Experience the situation.
Recognize it and identify a typical action for that situation.
Implement that action.
Retrieve, evaluate, and modify an action
Experience the situation.
Recognize it and identify a typical action for that situation.
Evaluate that action via mental simulation.
Until it seems likely to work in its familiar form, modify it and evaluate it again.
When satisfied, implement it.
Make sense of the situation and consider more than one action
Experience the situation.
Try to make sense of the situation until you have identified it as matching a familiar pattern.
Generate one or more plausible actions for this type of situation.
Evaluate each action via mental simulation, modifying it if necessary, until you have found one that seems likely to work.
Implement the selected action.

Source: Summarized on the basis of Klein, G. 1998. *Sources of Power: How People Make Decisions*. Cambridge, MA: MIT Press, Chapter 3.

a small fraction of cases, the decision makers really did find it necessary to consider (and perhaps modify) two or more alternative options before arriving at a satisfactory course of action. Some of the characteristics that make recognition-primed decision making ecologically rational are (1) the decision maker has a great deal of experience with previous similar situations; and (2) there is no time available for exhaustive generation and comparison of the alternative options.

These two conditions often apply to computer users as well, though the time pressure is usually not due to a dynamically changing emergency situation but rather to a need to proceed briskly with the activities that really interest the user.

5.7.2 Coherent Arbitrariness

A different line of research that revealed a striking tendency of people to repeat previous choices was conducted by Ariely (see, e.g., Ariely, Loewenstein, and Prelec 2006; Ariely 2008, Chapter 2).* In one typical experiment, Ariely asked study participants to state how much money they would want to be paid to endure unfamiliar unpleasant sounds of various durations. This choice problem was used because a participant's choice was bound to be largely arbitrary: Because people have no previous experience in paying money to avoid unpleasant sounds of this sort, there is no a priori notion of what a reasonable amount might be. And indeed, it was found that participants differed in the amounts they required and that their requirements could be influenced strongly by the manipulation of asking them about a particular price at the beginning ("Would you be willing to endure this sound for $.10/for $.50?").† But despite this arbitrariness, the participants' payment requirements were *coherent*: If one of them required a given amount of money to endure 10 seconds of a sound, they would require predictably larger amounts to endure 30 seconds or 60 seconds of the same sound. Evidently, participants were inclined to state their requirements in a way that was consistent with whatever requirement they had specified initially. The impact of the manipulation of the initial level could still be detected even after participants had received a good deal of relevant new information (e.g., information about the requirements of other participants).

One way of viewing coherent arbitrariness is as a result of reusing previous choices so as not to bother having to make the same choice over again. But it can also be seen as a reflection of people's desire to exhibit a consistent pattern of choices (see, e.g., Cialdini 2007, for a discussion of the ways in which *compliance professionals* such as salespersons exploit this tendency).

5.7.3 Choices Based on Habit

The most familiar way in which people repeat previous choices is when they act out of habit. The topic of habits is one of the oldest in psychology, but it continues to be an active area of research, bringing forth new theoretical perspectives, many of which now make use of neuropsychological concepts and research methods (see, e.g., Fu and Anderson 2006; Bayley, Frascino, and Squire 2005). For HCI researchers, a useful current synthesis is found in an article by Wood and Neal (2007, p. 813), who characterize habits as follows: "Habits are learned dispositions to repeat past responses. They are triggered by features of the context that have covaried frequently with past performance, including performance locations, preceding actions in a sequence, and particular people. Contexts activate habitual responses directly, without the mediation of goal states."

Although the ability to be triggered independently of any particular goal is a characteristic feature, habits can also interact with goals in various ways (Table 5.5), which are of particular interest to interaction designers who wish to take into account—or influence—habit-based behavior. The ways in which goals control habits are relevant to attempts to help users form appropriate habits or to leverage habits that they already have. The ways in which habits can conflict with goals are relevant, for example, to attempts to induce users not to act in accordance with an existing habit.

5.7.4 The Role of Skill Acquisition

Yet another reason why people often repeat previously made choices was already mentioned in the discussion of choice bracketing (Section 5.6.2): Suppose a user can choose between

* Paradoxically, Ariely introduced the term *coherent arbitrariness* in the first publication and switched to *arbitrary coherence* in the 2008 book.

† The provision of an *anchor* in this way is a frequent experimental method for influencing a judgment. Epley (2004) discusses the psychological mechanisms that underlie anchoring effects.

TABLE 5.5
Forms of Interaction between Goals and Habits

Relationship between Goals and Habits	Example
1. *Goals control habits*	
A person may intentionally form a habit.	"I'll back up my computer every evening just before leaving the office, so as to get into the habit of backing it up once a day …"
A person's goal-directed behavior may lead to the formation of a habit, without the person having any such intention.	"I decided on several days in a row to start my day by checking Facebook messages; and it became a bad habit."
A person can intentionally put themselves into a position where their choices can be made on the basis of habits.	"I make a point of shopping online only at my favorite web-based store: Once I've decided what to buy, I can go through the procedure of ordering the product without any attention or difficult decision making."
2. *Habits give rise to (inferences about) goals*	
A person can observe their own habitual behavior and make inferences about their own goals.	"I guess I assign high priority to good spelling and grammar: I always check the language of every e-mail message carefully before sending it off."
These inferences can in turn give rise to new goals.	"… So I guess I should spend more time proofreading my scientific articles before submitting them."
3. *Habits can conflict with goals*	
A person is sometimes aware that some habitual behavior of theirs conflicts with a goal that they have.	"I really have more important things to do at the beginning of each day than checking my Facebook messages."
But this awareness is not in itself enough to overcome the habitual behavior; two strategies are often successful:	
A. Actively and effortfully resist performing the undesired habitual response.	"I'm going to ignore the Facebook notification that just arrived!"
B. Change the situation so that you are no longer exposed to the cues that trigger the behavior.	Disable automatic notification about incoming Facebook messages; disable your entire Facebook account.

Source: Formulated on the Basis of Figure 1 of Wood, W., and D. T. Neal. 2007. *Psychol Rev* 114(4):843–63.

two ways (*A* and *B*) of performing a particular task, both of which seem about equally desirable at first (e.g., two different search engines for executing a web search; two alternative websites for downloading software; using the touch-pad or the isometric joystick on a new laptop). Even if the user's initial choice of *A* is essentially arbitrary, after executing *A*, the user will have become a bit more skilled at using *A*. So the next time, basically the same choice comes up, *A* should in principle be more attractive in terms of the user's skill at executing it. The user who engages in broad bracketing can anticipate this skill acquisition and take it into account when making the initial decision. But even a user who does not think that far ahead may notice the additional advantage of *A* after having chosen it at least once.

5.7.5 Example from Research on Method Selection

The importance of reusing previous choices was discussed in an influential article by Carroll and Rosson (1987) on the problem of method selection (Table 5.2). The authors began with the observation that computer users often persist in employing a relatively inefficient method to perform a given task even when they have more efficient methods available. One of the two explanations that the authors offered was *assimilation bias*: The authors noted that, if users can immediately think of an adequate method for performing a given task, they may use that method instead of taking the trouble to search for a better method. Assimilation bias is consistent with all four of the forms of repetition of previous choices discussed in Sections 5.7.1 through 5.7.4.*

5.7.6 Concluding Remarks on the Reuse of Past Choices

This section has shown that there are several different ways in which what a person chooses now can influence what they will choose in the future: Today's action can serve tomorrow as an example of a successful action or as a precedent; it can strengthen a habit or increase a person's skill. A negative implication is that an inappropriate choice can have more serious negative effects in the future than one would intuitively expect. The positive side is that, by supporting or influencing the user's actions in the short term, an interaction designer can increase the likelihood of appropriate choices in the longer term as well.

5.8 SOCIAL INFLUENCE

Another important general alternative (or complement) to consequentialist decision making is to be guided by the social context—specifically the examples, norms, and expectations

* The second explanation offered by Carroll and Rosson (1987), a production bias, can be seen as another example of the role of the user's current goal: the goal of getting the current job done is usually more important than the goal of increasing skill at using the system.

established by other people and the advice that they explicitly give.* For example, a person who has acquired a new computer for home use may consider at length what applications to install, what privacy and security settings to choose, and how to communicate with friends. When the same person works at the office, many of these decisions are likely to be influenced by written or unwritten rules, conventions, or social examples.

March (1994) offers a deep discussion of the view of decision making as *rule following*, which he contrasts with consequentialist decision making. In a similar vein, many authors in the HCI field have emphasized the importance of social and organizational context in influencing users' behavior (see, e.g., Button 2003). The point of view taken in this chapter is that social context accounts for some of the many considerations that are involved in decision making by an individual. In particular, a carefully selected presentation of aspects of the social context to an individual user can support or influence that user's choices.

The fact that the social environment often exerts a powerful influence on people's choices and decisions is known from everyday experience, and the mechanisms of social influence have been analyzed thoroughly in theories from social psychology and sociology. The diverse perspectives are associated with different concepts and terminology. The summary in Table 5.6 summarizes some commonly accepted ideas in everyday terms.

Note that, except for the final one, all of these considerations have something to do with consequences, either social or nonsocial. But when making a specific choice, a person may simply follow the general pattern of conforming to examples and expectations, without wondering about any associated consequences.

When it comes to interaction design and providing information to users, one general strategy is to provide users with more accurate or useful information about social examples and norms. The widely used paradigm of *collaborative filtering* for recommender systems (Jannach et al. 2011; Ricci et al. 2010) can be seen as providing information about choices that like-minded people have made. Most straightforwardly, this type of information serves the first function listed in Table 5.6. One of the relatively few applications of collaborative filtering to the support of choices about computer use, for the recommendation of commands, was presented recently by Li et al. (2011) (discussed in Chapter 20 of this handbook).

TABLE 5.6
Reasons Why People Can Be Influenced by Social Examples, Expectations, and Norms

Reason to choose in accordance with social influence	Example: using the company's social network
If others set an example (without necessarily expecting you to follow it):	
Their experience is a useful source of information.	"If these coworkers have acquired experience with this social network and are still using it, their experience must have been positive."
You want to enjoy practical benefits of conformity.	"There will be direct practical benefits to being in the same network as my coworkers, such as being able to exchange information with them conveniently."
You want to feel that you belong to their group.	"If I use the social network, I will feel more like a typical employee of this company."
If others expect you to make a particular choice:	
They can reward or punish you.	"If I don't use it, I may be subject to disapproval or even concrete disadvantages."
They have a legitimate reason for their expectation.	"The managers in my company have a right to expect me to do things like this."

The provision of extensive information about choices of other users is a typical feature of Web 2.0. Many online communities provide explicit information about the contribution behavior of their members, which can influence the contribution behavior of other members in several of the ways listed in Table 5.6.

These practices suggest that social information could be leveraged more extensively for the support of preferential choices and decisions about computer use—for example, to follow up belatedly on the observation made by Mackay (1991, p. 159) on the basis of her study of customization that "users want information about their own use and that of other people with similar job responsibilities and attitudes [on] which they can base their customization decisions."

5.9 LEARNING FROM EXPERIENCE

Especially when we consider sequences of similar choices that are made repeatedly, which is a typical HCI case, it becomes clear that an important aspect of choice and decision processes is what happens *after* the user has selected an option and experienced (to some extent) the consequences of a choice (see, e.g., Newell, Lagnado, and Shanks 2007). Aspects of learning have already been mentioned at various points in Section 5.7.

The model of action introduced by Norman (1986), which is well known in the HCI field, is worth bearing in mind in this context, even though it was not specifically intended to illuminate processes of preferential choice. In his discussion

* Many choices are made by a group of people rather than by an individual, as when a group of collaborating authors decides what text processing system to use to prepare their joint article. Group decision making in general involves some processes, such as interpersonal negotiation of compromises in cases of conflict of interest, which are not found in individual decision making (see, e.g., Kameda, Tindale, and Davis 2003, and Sorkin, Luan, and Itzkowitz 2004, for general treatments of group decision making; and Jameson and Smyth 2007, for a discussion of the special characteristics of recommender systems that make recommendations to groups). Although group decision making about computer use appears to be growing in importance with the increasing interconnectedness of computer users, the topic is omitted from this chapter for reasons of space.

of the *gulf of evaluation*, Norman distinguishes the phases of perceiving, interpreting, and evaluating the results of an action. Each of these phases can be seen as a way in which a chooser may have difficulty in learning from experience in making a certain type of choice. For example, a person who has acted on a decision to contribute one paragraph to a Wikipedia article will probably never know how many people have read the paragraph or how much they benefitted from it. The author may well notice the changes that other Wikipedia contributors make to the paragraph, but he may interpret them unrealistically and thereby arrive at an inappropriate evaluation of his original decision to contribute the paragraph.

Another example comes from the area of research on method selection: Bhavnani and John (2000) studied in depth expert users of computer-aided design systems who persisted in using inefficient methods: Among other things, they tended not to take advantage of the opportunity that their systems offered to perform an operation on multiple objects at one time. For example, when they needed to create three identical objects, they would draw them separately, instead of drawing one object and making two copies of it. One of the authors' explanations for the persistent use of inefficient methods concerned the fact that the users did not obtain clear feedback that revealed the inefficiency: The quality of the resulting drawings was in general identical, and the difference in execution times was not easy to notice from experience, especially if the users never tried the more efficient method in the first place. In view of this and other obstacles to spontaneous learning of the more efficient procedures, Bhavnani and his collaborators concluded that explicit training was required (see, e.g., Bhavnani, Peck, and Reif 2008).

In contrast, Gray and Boehm-Davis (2000) showed that, under more favorable learning conditions, users can sometimes take into account a difference between alternative *microstrategies* that involves only milliseconds of execution time. It can be seen, then, that the exact nature of the feedback that users receive about their choices can be crucial in determining whether preferential choices will improve on the basis of experience.

A recent trend in laboratory research on judgment and decision making (see, e.g., Rakow and Newell 2010) is to study *experienced-based choice*, where a person's choices about typical experimental problems such as pairs of gambles are based on concrete experience with the problems in question rather than on descriptions of the problems. For example, instead of being told that Option *A* offers a 10% chance of winning $12, whereas Option *B* guarantees a win of $1, a participant is allowed to click repeatedly on two buttons corresponding to the two options and observe the resulting rewards. An important issue in this sort of situation is the tension between *exploration* and *exploitation*: In order to learn efficiently which of the two options is preferable, a chooser should in principle systematically "explore" both of them, trying them out until it is clear which one is better—a process that may take some time, as in the example just given. But in practice, once a chooser has the impression, say, that Option *B* is better, there is a temptation to "exploit" this insight by consistently choosing *B*. The *production bias* observed by Carroll and Rosson (1987) can be interpreted in part as a result of users assigning higher priority to exploitation than to exploration.

Another typical obstacle is the difficulty of learning from one's own everyday experience very low probabilities such as those of a major hard disk failure, identity theft due to inadequate security measures, or an accident due to texting while driving.

As was mentioned in connection with choice bracketing (Section 5.6.2), one obstacle with broad bracketing is that it can be difficult for the chooser to learn from experience which of the broadly bracketed options yields the best results.

In cases like these, in which individual learning from experience faces serious obstacles, sources of guidance such as norms, policies, and the behavior of similar other persons play an especially important role. These cases also offer opportunities for interaction designers to improve choice and decision making noticeably by identifying the learning difficulty and taking steps to compensate for it.

5.10 CONCLUDING REMARKS

Readers who follow up on the references given in this chapter will discover many additional theoretical concepts, empirical results, and suggestive examples, including many on aspects of choice and decision making that could not be discussed in this chapter for reasons of space.* This literature can serve as a rich source of ideas about new ways to apply the HCI knowledge that is documented so thoroughly in the other chapters of this handbook.

ACKNOWLEDGMENTS

Preparation of this chapter benefited greatly from extensive discussions with Silvia Gabrielli, Per Ola Kristensson, Katharina Reinecke, Federica Cena, Cristina Gena, and Fabiana Vernero. Alan Dix supplied creative feedback and suggestions on several occasions. The author is grateful to the Handbook's editor, Julie Jacko, for her openness to the inclusion of a chapter on a new topic. The development of the perspective presented in the chapter was supported by the Autonomous Province of Trento in the context of the 2-year targeted research unit Prevolution. Preparation of the chapter itself was supported in part by the 7th Framework EU Integrating Project Glocal: *Event-based Retrieval of Networked Media* (http://www.glocal-project.eu/) under grant agreement 248984.

* For example, the collections edited by Koehler and Harvey (2004) and by Lichtenstein and Slovic (2006) include articles about the influence on decision making of affect and of culture.

REFERENCES

Ariely, D. 2008. *Predictably Irrational*. New York: HarperCollins.

Ariely, D., G. Loewenstein, and D. Prelec. 2006. “Coherent arbitrariness”: Stable demand curves without stable preferences. In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

Bagozzi, R. P. 2007. The legacy of the technology acceptance model and a proposal for a paradigm shift. *J Assoc Inf Syst* 8(4):243–55.

Bayley, P. J., J. C. Frascino, and L. R. Squire. 2005. Robust habit learning in the absence of awareness and independent of the medial temporal lobe. *Nature* 436(7050):550–3.

Bettman, J. R., M. F. Luce, and J. W. Payne. 2006. Constructive consumer choice processes. In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

Bhavnani, S. K., and B. E. John. 2000. The strategic use of complex computer systems. *Hum Comput Interact* 1(2/3):107–37.

Bhavnani, S. K., F. A. Peck, and F. Reif. 2008. Strategy-based instruction: Lessons learned in teaching the effective and efficient use of computer applications. *ACM Trans Comput Hum* Interact 15(1), Article 2.

Brooke, J. 1996. SUS—a quick and dirty usability scale. In *Usability Evaluation in Industry*, ed. P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, 189–94. London: Taylor & Francis.

Button, G. 2003. Studies of work in human-computer interaction. In *HCI Models, Theories, and Frameworks*, ed. J. M. Carroll. San Francisco, CA: Morgan Kaufmann.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Erlbaum.

Carroll, J. M., and M. B. Rosson. 1987. The paradox of the active user. In *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*, ed. J. M. Carroll, 80–111. Cambridge, MA: MIT Press.

Charman, S. C., and A. Howes. 2003. The adaptive user: An investigation into the cognitive and task constraints on the generation of new methods. *J Exp Psychol Appl* 9(4):236–48.

Cialdini, R. B. 2007. *Influence: The Psychology of Persuasion*. New York: HarperCollins.

Epley, N. 2004. A tale of tuned decks? Anchoring as accessibility and anchoring as adjustment. In *Blackwell Handbook of Judgment and Decision Making*, ed. D. J. Koehler and N. Harvey. Malden, MA: Blackwell.

Fishbein, M., and I. Ajzen. 2010. *Predicting and Changing Behavior: The Reasoned Action Approach*. New York: Taylor & Francis.

Fogg, B. J. 2003. *Persuasive Technology: Using Computers to Change What We Think and Do*. San Francisco, CA: Morgan Kaufmann.

Fogg, B. J., G. Cueller, and D. Danielson. 2008. Motivating, influencing, and persuading users: An introduction to captology. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, ed. A. Sears and J. A. Jacko, 2nd ed., 133–46. Boca Raton, FL: CRC Press.

Fu, W., and J. R. Anderson. 2006. From recurrent choice to skill learning: A reinforcement-learning model. *J Exp Psychol Gen* 135(2):184–206.

Gabrielli, S., and A. Jameson. 2009. Obstacles to option setting: Initial results with a heuristic walkthrough method. In *Human-Computer Interaction—INTERACT 2009, 12th IFIP TC 13 International Conference*, ed. T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. Prates, and M. Winckler, 400–3. Berlin: Springer.

Gardiner, M. M., and B. Christie, eds. 1987. *Applying Cognitive Psychology to User-Interface Design*. Chichester, England: Wiley.

Gigerenzer, G. 2007. *Gut Feelings: The Intelligence of the Unconscious*. London: Penguin.

Gigerenzer, G., and P. M. Todd, eds. 1999. *Simple Heuristics That Make us Smart*. New York: Oxford.

Gray, W. D., and D. A. Boehm-Davis. 2000. Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *J Exp Psychol Appl* 6(4):322–35.

Hastie, R. 2001. Problems for judgment and decision making. *Annu Rev Psychol* 52:653–83.

Hastie, R., and R. M. Dawes. 2010. *Rational Choice in an Uncertain World*. Thousand Oaks, CA: Sage.

Iachello, G., and J. Hong. 2007. End-user privacy in human-computer interaction. *Found Trends Hum Comput Interact* 1(1):1–137.

Iyengar, S. 2010. *The Art of Choosing*. New York: Hachette.

Jameson, A., S. Gabrielli, P. O. Kristensson, K. Reinecke, C. Gena, F. Cena, and F. Vernero. 2011. How can we support users’ preferential choice? In *Extended Abstracts of the 2011 Conference on Human Factors in Computing Systems,* ed. D. Tan, B. Begole, and W. A. Kellog. New York: ACM.

Jameson, A., and K. Klöckner. 2005. User multitasking with mobile multimodal systems. In *Spoken Multimodal Human-Computer Dialogue in Mobile Environments*, ed. W. Minker, D. Bühler, and L. Dybkjær, 349–77. Dordrecht, The Netherlands: Springer.

Jameson, A., and B. Smyth. 2007. Recommendation to groups. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 596–627. Berlin, Germany: Springer.

Jannach, D., M. Zanker, A. Felfernig, and G. Friedrich. 2011. *Recommender Systems: An Introduction*. Cambridge, UK: Cambridge.

Johnson, J. 2010. *Designing with the Mind in Mind: A Simple Guide to Understanding User Interface Design Rules*. Burlington, MA: Morgan Kaufmann.

Johnson, E. J., and D. G. Goldstein. 2006. Do defaults save lives? In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

Kahneman, D., and A. Tversky. 1979. Prospect theory: An analysis of decision under risk. *Econometrica* 47(2):263–95.

Kameda, T., R. S. Tindale, and J. H. Davis. 2003. Cognitions, preferences, and social sharedness: Past, present, and future directions in group decision making. In *Emerging Perspectives on Judgment and Decision Research*, ed. S. L. Schneider and J. Shanteau. Cambridge, UK: Cambridge University Press.

Keeney, R. L. 1992. *Value-Focused Thinking: A Path to Creative Decisionmaking*. Cambridge, MA: Harvard.

Kieras, D. 2008. Model-based evaluation. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, ed. A. Sears and J. A. Jacko, 2nd ed., 1191–208. Boca Raton, FL: CRC Press.

Klein, G. 1998. *Sources of Power: How People Make Decisions*. Cambridge, MA: MIT Press.

Klein, G. 2008. Naturalistic decision making. *Hum Factors* 50(3):456–60.

Koehler, D. J., and N. Harvey, eds. 2004. *Blackwell Handbook of Judgment and Decision Making*. Malden, MA: Blackwell.

Kuniavsky, M. 2010. *Smart Things: Ubiquitous Computing User Experience Design*. Burlington, MA: Morgan Kaufmann.

Law, E. L., V. Roto, M. Hassenzahl, A. P. Vermeeren, and J. Kort. 2009. Understanding, scoping and defining user experience: A survey approach. In *Human Factors in Computing Systems: CHI 2009 Conference Proceedings*, ed. S. Greenberg, S. Hudson, K. Hinckley, M. R. Morris, and D. R. Olsen, 719–28. New York: ACM.

Lerner, J. S., and P. E. Tetlock. 2003. Bridging individual, interpersonal, and institutional approaches to judgment and decision making: The impact of accountability on cognitive bias. In *Emerging Perspectives on Judgment and Decision Research*, ed. S. L. Schneider and J. Shanteau. Cambridge, UK: Cambridge University Press.

Li, W., J. Matejka, T. Grossman, J. Konstan, and G. Fitzmaurice. 2011. Design and evaluation of a command recommendation system for software applications. *ACM Trans Comput Hum Interact* 18(2), Article 6.

Lichtenstein, S., and P. Slovic, eds. 2006. *The Construction of Preference*. Cambridge, UK: Cambridge University Press.

Loewenstein, G., and J. Elster, eds. 1992. *Choice over Time*. New York: Sage.

Loewenstein, G., D. Read, and R. Baumeister, eds. 2003. *Time and Decision*. New York: Sage.

Loraas, T., and M. C. Diaz. 2009. Learning new uses of technology: Situational goal orientation matters. *Int J Hum Comput Stud* 67:50–61.

Mackay, W. E. 1991. Triggers and barriers to customizing software, In *Human Factors in Computing Systems: CHI 1991 Conference Proceedings*, ed. S. P. Robertson, G. M. Olson, and J. S. Olson, 153–60. New York: ACM.

Mandel, N., and E. J. Johnson. 2006. When web pages influence choice: Effects of visual primes on experts and novices. In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

March, J. G. 1994. *A Primer on Decision Making: How Decisions Happen*. New York: The Free Press.

Maule, A. J. 2010. Can computers help overcome limitations in human decision making? *Int J Hum Comput Interact* 26(2–3):108–19.

McDowell, L., O. Etzioni, S. D. Gribble, A. Halevy, H. Levy, W. Pentney, D. Verma, and S. Vlasseva. 2003. Mangrove: Enticing ordinary people onto the semantic web via instant gratification. *Proceedings of ISWC 2003*, 754–70. Sanibel Island, Florida. Berlin: Springer.

McGrenere, J., R. M. Baecker, and K. S. Booth. 2007. A field evaluation of an adaptable two-interface design for feature-rich software. *ACM Trans Comput Hum Interact* 14(1), Article 3.

Newell, B. R., D. A. Lagnado, and D. R. Shanks. 2007. *Straight Choices: The Psychology of Decision Making*. Hove, UK: Psychology Press.

Norman, D. A. 1986. Cognitive engineering. In *User Centered System Design: New Perspectives on Human-Computer Interaction*, ed. D. A. Norman and S. W. Draper, 31–61. Hillsdale, NJ: Erlbaum.

Rachlin, H. 2000. *The Science of Self-Control*. Cambridge, MA: Harvard.

Rakow, T., and B. R. Newell. 2010. Degrees of uncertainty: An overview and framework for future research on experience-based choice. *J Behav Decis Mak* 23:1–14.

Read, D. 2004. Intertemporal choice. In *Blackwell Handbook of Judgment and Decision Making*, ed. D. J. Koehler and N. Harvey. Malden, MA: Blackwell.

Read, D., G. Loewenstein, and M. Rabin. 2006. Choice bracketing. In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

Ricci, F., L. Rokach, B. Shapira, and P. B. Kantor, eds. 2010. *Recommender Systems Handbook*. Berlin, Germany: Springer.

Schneider, S. L., and M. D. Barnes. 2003. What do people really want? Goals and context in decision making. In *Emerging Perspectives on Judgment and Decision Research*, ed. S. L. Schneider and J. Shanteau. Cambridge, UK: Cambridge University Press.

Schneider, S. L., and J. Shanteau, eds. 2003. *Emerging Perspectives on Judgment and Decision Research*. Cambridge, UK: Cambridge University Press.

Shafir, E., I. Simonson, and A. Tversky. 2006. Reason-based choice. In *The Construction of Preference*, ed. S. Lichtenstein and P. Slovic. Cambridge, UK: Cambridge University Press.

Sorkin, R. D., S. Luan, and J. Itzkowitz. 2004. Group decision and deliberation: A distributed detection process. In *Blackwell Handbook of Judgment and Decision Making*, ed. D. J. Koehler and N. Harvey. Malden, MA: Blackwell.

Thaler, R. H., and C. R. Sunstein. 2008. *Nudge: Improving Decisions About Health, Wealth, and Happiness*. New Haven, CT: Yale University Press.

Tintarev, N., and J. Masthoff. 2010. Explanation of recommendations. In *Recommender Systems Handbook*, ed. F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. Berlin, Germany: Springer.

Venkatesh, V., and F. Davis. 2000. A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Manag Sci* 46(2):186–204.

Venkatesh, V., M. G. Morris, G. B. Davis, and F. D. Davis. 2003. User acceptance of information technology: Toward a unified view. *MIS Quart* 27(346):425–78.

Weber, E. U., and E. J. Johnson. 2009. Mindful judgment and decision making. *Annu Rev Psychol* 60:53–88.

Wickens, C. D., and J. G. Hollands. 2000. *Engineering Psychology and Human Performance*. Upper Saddle River, NJ: Prentice Hall.

Wilson, T. D. 2002. *Strangers to Ourselves: Discovering the Adaptive Unconscious*. Cambridge, MA: Harvard.

Wood, W., and D. T. Neal. 2007. A new look at habits and the habit-goal interface. *Psychol Rev* 114(4):843–63.

Yates, J. F., E. S. Veinott, and A. L. Patalano. 2003. Hard decisions, bad decisions: On decision quality and decision aiding, In *Emerging Perspectives on Judgment and Decision Research*, ed. S. L. Schneider and J. Shanteau. Cambridge, UK: Cambridge University Press.

Young, R. M., and A. MacLean. 1988. Choosing between methods: Analysing the user's decision space in terms of schemas and linear models. In *Human Factors in Computing Systems: CHI 1988 Conference Proceedings*, ed. J. J. O'Hare, 139–43. New York: ACM.

Part II

Computers in HCI

6 Input Technologies and Techniques

Ken Hinckley and Daniel Wigdor

CONTENTS

6.1 INTRODUCTION: A SHIFTING LANDSCAPE

The barometer has fallen and a brisk wind blows. Input and interaction are not what they used to be. Gusts drift over the line in the sand that separates input from output. The academic division between input and output as topics of study was never a principled one to begin with. But this is perhaps clearer now than ever before: Many people now primarily experience computing by direct-touch input on their displays. The shifting sands are rapidly unearthing a future where any surface on which we can display information will also serve as a surface that sees, hears, feels, and otherwise senses all our interactions. As some would have it, the encroaching dunes have all but buried the mouse, keyboard, and other indirect-input devices as archeological curiosities to be unearthed by a future generation of researchers, explorers, and scoundrels.

What is the prospector of innovative devices and interaction techniques to make of this new landscape? Is it a desert or is it an oasis? In our view it is a little bit of each. Everyone is now familiar with multitouch as the defining example of direct interaction, but it is not the whole story. As our colleague Bill Buxton constantly likes to remind us, a key lesson to remember is the following: *Everything, including touch, is best for something and worst for something else*.

The goal of this chapter is to help you to understand why, how, and under what circumstances, a given input modality is most appropriate. This will be done by way of a survey of illustrative examples, devices, techniques, and conceptual tools. It will also help you to understand why direct interaction is about much more than just multitouch. Direct interaction includes not only many variations on touch itself but also modalities such as pen input, motion and postural sensing, and proximal and spatial interactions in the physical space beyond the display. Each of these in turn exhibits its own idiosyncratic strengths and weaknesses.

6.1.1 Every Coin Has Two Sides

Having said that every coin has two sides, is it even desirable to think of devices and modalities in terms of strengths and weaknesses? Perhaps not, and indeed we encourage the reader to move beyond this categorical mind-set. Input modalities share a number of common properties. If a designer understands these properties thoroughly, then nearly any property of an input device can be turned to one's advantage when used appropriately in an interface design.

For example, take the property of *mechanical intermediary*. A modality either requires a mechanical intermediary, such as a pen, or it does not, as is the case with direct-touch input. Think of this property of mechanical intermediary as a coin with two sides: Heads requires an intermediary, but tails does not. Which side of the coin should we choose? Well, of course, it depends.

If we choose heads, then *the pen* has an advantage because it employs a mechanical intermediary. A stylus is a familiar tool to which users may bring highly developed skills for handwriting, sketching, and drawing. The body of the pen provides a lever arm that affords a tripod grip for precise control, and its tapered tip enables one to indicate small objects on the screen. It also affords a purchase for secondary controls such as buttons, an eraser head, or perhaps a trigger (as in an airbrush, e.g., which modulates the flow of paint). A skilled interaction designer can leverage these attributes to produce compelling user experiences.

If we choose tails instead, on the other side of the coin, then touch has an advantage because it does not require a mechanical intermediary. Unlike a stylus, there is nothing to lose. The user always carries their finger with them. Furthermore, the user can start interacting immediately. There is no additional *acquisition time* required to grasp or unsheathe a stylus. This becomes a critical factor in mobile interaction, where interacting with the real world and other individuals is the user's primary task, forcing users to wedge their interactions with a device into ever-narrowing fragments of time and attention.

That is, the property of mechanical intermediary shared by both the pen and touch input modalities is neither inherently good nor inherently bad. It is just a property. Correctly choosing heads or tails depends on what the user is trying to accomplish, as well as how that task fits into the larger set of activities that a designer seeks to support with an interface. The *exact same property* of a modality that offers a marked advantage for one task, or context of use, turns about-face and becomes a crippling liability in another task. In our example, the requirement of a mechanical intermediary is both a strength and a weakness for the pen, as is the lack of an intermediary for touch. But having these two opposing sides makes the coin no less valuable. Indeed, our ability to trade off these two opposing facets against one another is precisely what gives the coin its value in the currency of design.

Hopefully by now the lesson is clear. The designer should not just extol the virtues of a particular modality. The designer does not truly understand an input until he or she can articulate its shortcomings just as thoroughly as he or she can articulate its advantages. The emerging trend of user experiences that integrate both pen and touch inputs (as well as other modalities) illustrates the practical need for this perspective (Hinckley et al. 2010; Zeleznik, Bragdon, Ko et al. 2010; Frisch, Heydekorn, and Dachselt 2009; Brandl et al. 2008).

6.1.2 From a Competitive to a Cooperative Landscape of Devices

Let us now step away from the computer for a moment and look at the example of handwriting with pencil and paper. These are "devices" that one interacts with in the real world, after all. Consider the simple question: *Which hand do you write with, right or left?* When we give talks on this subject, we have the audience raise their hands. Who uses their right hand? Do we have any left-handers in the audience? Which hand would you, the reader, raise? Now, of course, we have led you into a trap because *you are all wrong. No matter which hand you raised, you are wrong.* This is not a trick question. Rather, the question is fundamentally ill posed. People write with both hands, as demonstrated by Guiard (1987).

What Figure 6.1 shows is the result of writing on a sheet of paper. On the right, we see the impressions left by the pen on a sheet of transfer paper surreptitiously left underneath. That is, it records the movements of the pen relative to the desk rather than relative to the paper, and it reveals that the nonpreferred hand dynamically adjusts the position of the paper to suit the action of the preferred hand. Hence, both hands play a role in handwriting, but each hand plays a *specialized role* that is *complementary* to the other hand. In the same way that one's hands are specialized, we can think of input modalities and devices as taking on specialized roles in a larger ecosystem of interaction.

In both popular press and research papers, one often sees input modalities framed in *competitive* terms. How many times have we seen headlines proclaiming that the next magical device will make the mouse and keyboard obsolete? How many studies have we seen that compare device A with device B for some task? This implies a winner, as in a football match, where either A or B achieves victory. These type of headlines and studies beg research questions, even if only implicitly, of the following form: *Which is better, touch or stylus (or the mouse, or motion sensing, or freehand gesturing, or name your favorite input modality, device, or interaction technique)?*

If we find ourselves asking a question of this sort, then we must once again recognize it as an ill-posed query. We are not saying that studies that compare individual techniques cannot be illuminating. They can be illuminating if done in a principled manner that carefully assesses and controls the factors underlying performance, rather than just comparing two arbitrary techniques. However, isolated studies cannot be the only things that we publish nor should they be framed

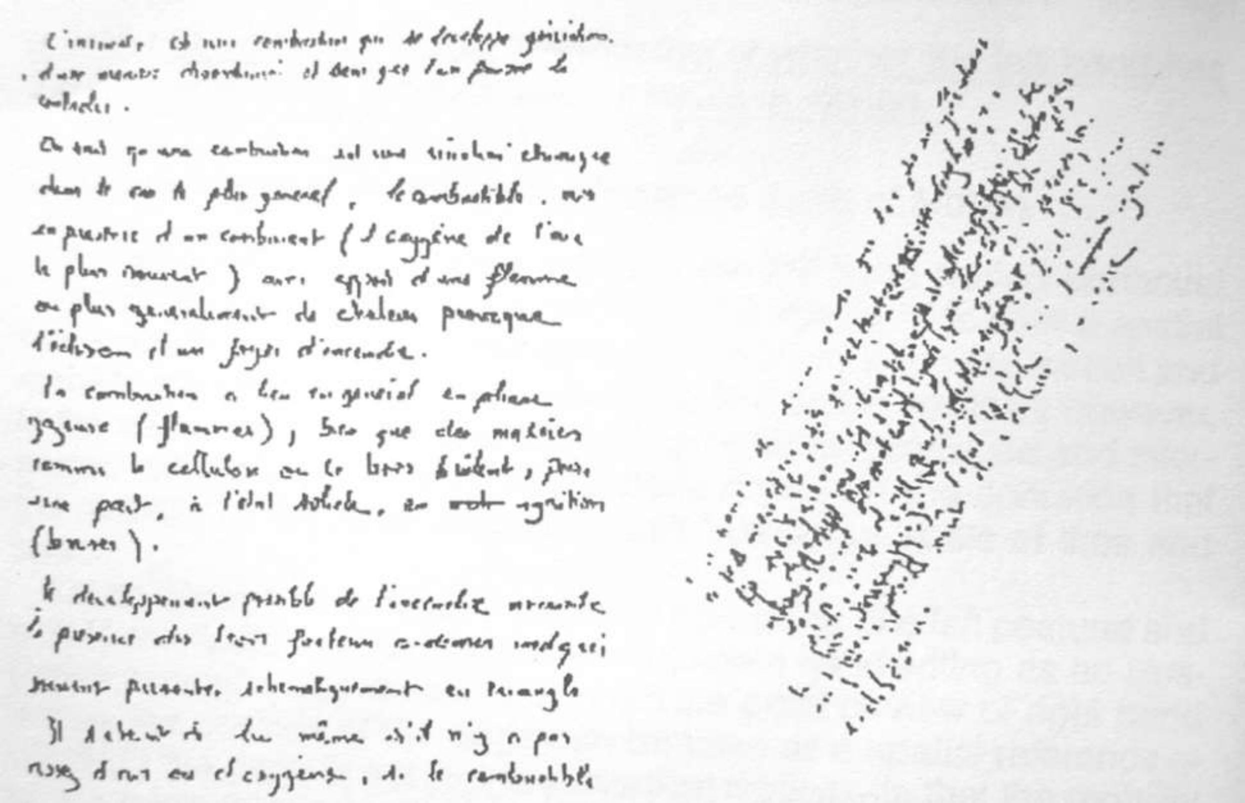


FIGURE 6.1 Guiard's transfer paper experiment, with the full sheet of paper shown on the left and the impressions left on an underlying sheet of transfer paper shown on the right. This shows how the nonpreferred hand orients and shifts the sheet of paper to suit the action of the preferred hand while writing. (From Guiard, Y., *J Motor Behav*, 19(4):486–517, 1987. http://www.informaworld.com. With permission.)

and extrapolated without due consideration of the ecology of devices and techniques in which they reside.

Following Guiard (1987), we believe the correct question to ask is one that takes a *cooperative* perspective, as well as a broad view of input modalities and form factors. There are multiple modalities and devices. We should not try to do everything with any one of them. We should instead seek to understand how input modalities and techniques can complement one another such that the advantages of one make up for the shortcomings of another.

Interactive system design should encourage a division of labor between a collection of input devices, sensing modalities, and interaction techniques that together sum to more than the whole of the parts. We should seek out logical design principles (or rules of thumb) that help to guide the ways in which we combine techniques. That is, we should frame our input research and system designs around questions of the following form: *What is the logic of the division of labor between touch, pen, motion and postural sensing, proximal and spatial interactions beyond the display, and a diverse ecology of devices and form factors?*

If we can succeed in bringing about this shift in perspective, then perhaps this chapter can form a more enduring survey of the terrain than the gold rush that the industry is witnessing to do anything and everything with touch, and with touch alone.

6.1.3 Multiple-Modality Perspective

The discussion in Section 6.1.2 implies that the reader should take a multiple-modality perspective on the design of interactive systems. The Apple iPhone has been lauded for its multitouch interface, but do not forget the additional modalities such as the use of proximity sensing to avoid “ear dialing” (Dietz and Yerazunis 2001) or the use of orientation sensing to automatically rotate the screen (Hinckley et al. 2000; Schmidt, Beigl, and Gellersen 1999), both of which enrich the experience offered by the device.

If we accept then that multitouch is likely the tip of the iceberg in terms of what is possible for a natural and compelling direct-input user experience, then the combination and infusion of touch with rich sensory information from other modalities represents the great frozen massif beneath the waterline in terms of future innovation and research contributions. Rather than focusing on one modality, such as touch, our collective goal as a community of researchers, designers, and practitioners should be to understand how to most effectively design systems that use input modalities in combination, where the strengths of one compensate for the limitations of another. When a system does not have to provide coverage of all possible functions with a single input modality, implicitly this leads one to ask where each modality should be used to the best advantage, where a particular modality should not be used, and in what cases modalities should perhaps be treated interchangeably (Hinckley et al. 2010).

Nonetheless, we must be pragmatic and recognize that at times an interface design must focus on a single modality. But even if your intention is to design a single-modality user experience, such as a touch-screen device, our belief is that you can design a better and more compelling experience if you do so with a broad perspective of the limitations of that modality, as well as good working knowledge of the capabilities of other modalities. Perhaps you can even anticipate future developments and dovetail your design to intersect with capabilities that other modalities may add to the user experience in future generations of a device or system, or enable richer capabilities in the presence of secondary sensors or peripherals that augment a system’s core capabilities.

6.1.4 Breadth–Depth Dichotomy

With respect to systems and devices, there are two basic design strategies: (1) We can design for breadth, supporting a wide ecosystem of hardware and modalities with a one-size-fits-all design. Designing for breadth creates a common platform that unifies many different devices and experiences, with all the benefits (as well as the drawbacks) that are inherent in an accretion of features and interfaces. (2) Or we can design for depth, with a core sample carefully drilled through the many strata of target market, user experience, input modality, display size and form factor, operating system, and application design. The market shows that designing for depth can yield simplification of user interfaces as well as compelling experiences. Yet the market has also shown that designing for breadth has immense value when done well. Which way to turn? We refer to this fundamental design dilemma as the *breadth–depth dichotomy*.

This dichotomy pervades every element of design as well as software and hardware development. Successfully addressing this tension is a key challenge for designers of software and of devices supporting direct interaction. It is as if we are lost at sea and are awaiting the arrival of an automated method to “convert” applications and experiences for different contexts of use, but none is apparent on the horizon. This chapter aims to provide the reader with a modest navigational tool to help find his or her way through this unsettled ocean of design decisions and their subtle implications.

Whether one adopts a unimodal or a multimodal perspective, excellence in user interface design requires tailoring an interface to the input method. This tailoring extends to form factor as well as modality. Multitouch interaction with a phone is often conducted with thumbs while holding the device. Multitouch interaction with wall-mounted displays is conducted with fingertips and by keeping the arms extended. The physicality of the interaction is clearly different for the two cases, as is the context of use; it is likely the tasks that the user will be performing are also different. As another example, many common websites and applications have both desktop and mobile editions. Students of human–computer interaction (HCI) should consider these and ask the following: *Does each have a different user interface model? What are the commonalities, and what are the differences?* The answer varies depending on the application, because no standard has yet been reached.

This further extends to a society of devices that users expect to work together with consistent experiences that are nonetheless tailored to the form factor. User interface consistency should not be viewed as a rigid construct but as a flexible one that maintains *consistency of user expectations* given the form factor, input modality, context of use, and the specific task at hand. Handheld devices, e-readers, tablets, booklets, slates, desktops, tabletops, laptops, and wall-mounted displays each have their own unique affordances, and each may be encountered by the same user in varying contexts. Thus, from the user's perspective, the most "consistent" user experience may in fact be one that is not consistent with respect to the specific actions that the user must articulate to use a given form factor.

6.1.5 Designing User Interfaces: The Loss of a Closely Held Abstraction

The widespread shift to direct interaction, in service of mobility as well as other concerns, has ushered in a storm that batters us on our voyage of discovery. We have arrived at a strange land lashed by impetuous gales that we must weather while avoiding the uncharted seamounts and rocky shoals that abound its wild coast. But in the lee of the tempest, islands of compelling new interactions, form factors, and user experiences can be discovered that tempt both the adventurous and the foolhardy to ply these treacherous waters. The result, certainly, will be many shipwrecks, and also some wondrous new systems and devices that were never before imagined.

Our voyage has led us far away from the Old World of the so-called WIMP (windows, icons, menus, and pointers) desktop user interface. This paradigm has been painstakingly designed over more than four decades for multimodal control with a keyboard and with a mouse, and was achieved in no small part through the cursor. The cursor is a logical abstraction of the user's locus of attention, known by many names (originally as the bug, and alternately as the telepointer, pointer, or tracking symbol). The abstract representation of the cursor serves as an intermediary that separates the concerns of input and output. Direct interaction does away with this separation and abstraction, leading us toward a world where each device requires specific and particular software design.

One must keep in mind that the WIMP graphical user interface (GUI) is highly optimized for the particular pairing of mouse and keyboard input devices. Although it is possible to control the pointer and enter text using other input devices, doing so is more difficult and less efficient than that with mouse and keyboard. Many modern platforms rely on the accoutrements of the WIMP interface (buttons, sliders, checkboxes, windows, and so forth), simultaneously forgetting that these are optimized for a pair of input devices that are not actually present on the systems supported by the platforms. In short, new modalities require new user interfaces and interaction techniques if one wishes to make the most of their capabilities.

To the extent that we succeed, this chapter forms a chart, which, despite being a rough guide riddled with terra incognitae, can assist the explorers of this new interactive world. Through this exercise, we seek to equip the explorer with what we believe is the most valuable tool of all: the confidence in knowing which questions to ask. This will be based on as deep an understanding of the properties of input devices as we are able to impart through this short chapter, and which will be rooted in perhaps the only indisputable fact in the ever-changing landscape of design: Everything is best for something and worst for something else.

Section 6.2.1 first considers, in a little more depth, what is actually meant (in our opinion) by "natural" user interfaces. We discuss terminology; enumerate some common properties of input devices; and provide examples of how these properties apply to familiar examples of devices such as touch screens, pens, and mice. However, our emphasis is more on the particular concerns of direct-input devices and less on indirect devices. We then turn our attention to state-transition models of devices and how these relate to interactive techniques, as well as the hierarchies of fundamental tasks that delineate the units of interaction in a user interface. We briefly examine a number of models and theories that are commonly used to evaluate human performance with interaction techniques and devices. We discuss how to transform input signals for use in applications and analyze some factors and considerations for the design of appropriate feedback in response to inputs. The chapter then takes a brief sojourn into the specialized realm of discrete symbolic entry, including mobile and keyboard-based text entry. In Sections 6.11 and 6.12, we consider some higher-level topics such as input modalities and general strategies for interaction that transcend particular input devices, as well as the impact on input techniques of a diverse and evolving ecosystem of form factors. We conclude with some thoughts about future trends and opportunities in this field.

6.2 UNDERSTANDING INPUT TECHNOLOGIES

Input devices sense physical properties of people, places, or things. But any treatment of input devices without considering corresponding visual feedback is like trying to use a pencil without paper. This chapter treats input technologies at the level of interaction techniques, which provide a way for users to accomplish tasks by combining input with appropriate feedback. An interaction designer must consider the physical sensor, feedback presented to the user, the ergonomic and industrial design of the device, and the interplay between all the interaction techniques supported by a system, as well as the interplay between a specific device and other devices in its surrounding digital ecology.

A designer who understands input technologies and the task requirements of users has a better chance of designing interaction techniques that match a user's natural workflow and that take appropriate advantage of a user's innate abilities. Making an optimal choice for tasks in isolation often leads to a poor design, so the designer must weigh competing design requirements as well as transitions between tasks.

6.2.1 Is This Input Method Natural?—Why This Is an Ill-Posed Question

The word natural can be applied to interactions with systems. A reasonable operational definition for a natural UI is that the experience of using a system matches expectations such that it is always clear to the user how to proceed and only a few steps (with a minimum of physical and cognitive effort) are required to complete common tasks. It is a common mistake to attribute the naturalness of a product to underlying input technology. A touch screen, or any other input method for that matter, is not inherently natural.

A common naive reaction to direct-touch and gestural input, for example, is to suggest that it is "more natural" than other input methods because interactions can be accomplished by hand gestures that correspond to movements that users naturally make as a result of their everyday real-world experiences. Wobbrock, Morris, and Wilson (2009) explored this hypothesis. They presented user-study participants with "before" and "after" images depicting a particular interaction, such as moving an object, duplicating an object, or changing the color of an object. They then asked the participant to perform the gesture they believed would create that change in the system. Three of their experimental tasks are summarized in Figure 6.2.

You can conduct this experiment on yourself: Given a multitouch screen, what gesture do you expect will transform Example 1 in Figure 6.2 from its before state to its after state? For Example 1, almost everyone suggests the same action as you likely just envisioned, that of dragging the square with a single finger.

For Example 2, results are more equivocal. You probably have to think a bit longer before deciding on an appropriate gesture. If we present this example to multiple users, there is some agreement across users, but there are also a number of different gestures that users choose to naturally indicate the duplication of an object.

For Example 3, all bets are off. There is almost no agreement across a set of users as to a natural manipulative gesture to change the color of an object.

What this exercise shows is that *there is no inherently natural set of gestures* for performing anything beyond a couple of the most commonplace multitouch manipulations. The gestures that users suggest for a manipulation can be used as a source of design inspiration, but this observation does not lead to a set of gestures that is consistent or inherently more natural than any other set.

6.2.2 Terminology: Input Device versus Interaction Technique versus Conceptual Model versus User Experience

We take a brief pause in this section to clarify the terminology we use. To some extent we can use devices, techniques, models, and experiences as interchangeable terms, but what is more important is for the designer to think about these things at different conceptual levels so as to consider a holistic view of interactive systems. The terminology used is as follows:

Input device: An input device is a transducer that senses physical properties of people, places, or things. What we normally conceive of as an input device (such as a mouse) is often a collection of transducers (Card, Mackinlay, and Robertson 1990, 1991), such as a relative (x, y) motion sensor, physical buttons, and a wheel for scrolling.

Conceptual model: A conceptual model is a coherent model that users visualize about the function of a system—what the system is, how it works, and how it will respond to users' input. Thus users can determine what input they need to give to the system to achieve a desired result. Conceptual models can be taught, but they are continually refined throughout a user's interaction with a system, given its responses to the user's input. For example, the core conceptual model of a GUI is the point-and-click metaphor of moving the tracking symbol on top of objects on the screen and then acting on them by clicking or dragging the mouse. Likewise, the desktop metaphor of folders and files is a higher-level conceptual model of how information is stored and retrieved in a computer.

Interaction technique: An interaction technique is the fusion of input and output, consisting of all hardware and software elements, that provides a way for a user to accomplish a task for a particular conceptual model. For example, pinch-to-zoom has become a de facto standard for touch-screen devices, just as clicking and dragging the mouse as a way to move items is a staple of the desktop metaphor. It is important to remember that the cursor itself is a single element of an interaction technique and that this is just one of many possible ways to interact using a relative pointing device. Interaction techniques typically

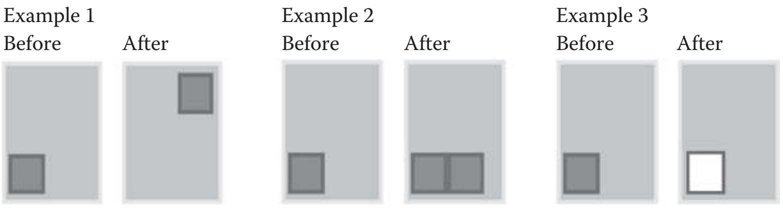


FIGURE 6.2 Before–after pairs for three example gestures that a user performs on a hypothetical multitouch system. What touch gesture would you expect transforms each example from its before state to its after state?

vary across input devices based on the strengths the device's sensing capabilities, the ability to readily incorporate *state transitions* such as button presses into the design of the device, and the user's physical abilities and hand comfort when using the device.

User interface: A user interface is the representation of a system—the summation of all its input devices, conceptual models, and interaction techniques—with which a user interacts. It is the responsibility of the user interface to represent and reinforce the user's conceptual model of the system in concert with the input device and interaction techniques, as well as presenting affordances and constraints that make it clear to the users how to achieve key tasks. User interfaces, despite being stereotyped as GUIs, also include auditory, tactile, and kinesthetic qualities, even if such secondary feedback results only from the passive mechanical feedback from physical input devices.

User experience: User experience is the broad array of outputs perceived and inputs given by a user when interacting with a user interface, as well as the higher-level goals, cognitive states, emotions, and social interactions that such experiences support and engender.

6.2.3 Input Device Properties

The variety of pointing devices available is bewildering, but a few common properties characterize the important characteristics of most input sensors. These properties help a designer in understanding a device and in anticipating potential problems. We will first consider these device properties in general and then show how they apply to some common input devices.

Property sensed: Most devices sense linear position, motion, or force; rotary devices sense angle, change in angle, and torque (Buxton 1995c; Card, Mackinlay, and Robertson 1991). For example, touch screens sense the position of one or more fingers, mice sense motion (change in position), and isometric joysticks sense force. The property sensed determines the most appropriate mapping from input to output, or transfer function, for a device. Position-sensing devices are absolute input devices, whereas motion-sensing devices are relative input devices. A relative device, such as the mouse, requires visual feedback in the form of a cursor to indicate a screen location.

States sensed: Direct-input devices, such as touch screens, touchpads, and pen input devices, are unique in that they sense not only position but also contact events (i.e., finger-down and finger-up events), an ability that traditional pointing devices lack (Hinckley and Sinclair 1999). Action of these devices cannot be treated as the equivalent of mouse-click events. This distinction is obvious, but its implications can be quite subtle in the design of interaction techniques even for a designer well versed in designing such devices. See Section 6.5 for further discussion.

Number of dimensions: Devices sense one or more input dimensions. For example, a mouse senses two linear dimensions of motion, a knob senses one angular dimension, and a magnetic tracker with six degrees of freedom measures three position dimensions and three orientation dimensions (Bowman et al. 2004; Hinckley et al. 1994a; Zhai 1998). A pair of knobs or a mouse with a scroll wheel senses separate input dimensions and thus forms a "1D + 1D" device (where 1D stands for one dimensional) or a "2D + 1D" (where 2D stands for two dimensional) multichannel device, respectively (Zhai, Smith, and Selker 1997).

Device acquisition time: The average time to move one's hand to a device is known as acquisition time. Homing time is the time to return from a device to a "home" position (e.g., return from mouse to keyboard). The effectiveness of a device for pointing tends to dominate acquisition time (Douglas and Mithal 1994). Thus, integration of a pointing device with the keyboard may not improve overall performance, but evaluations must still assess any influence of acquisition times (Dillon, Eday, and Tombaugh 1990; Hinckley et al. 2006) to be certain that a comparison between the devices is fair. The elimination of acquisition time is a key advantage of touch interaction for mobile interactions, but of course some homing time is still required to bring one's hand into contact with the screen.

Other metrics: System designers must weigh other performance metrics also (Card, Mackinlay, and Robertson 1990), including pointing speed and accuracy, error rates, learning time, footprint and gain (Accot and Zhai 2001; MacKenzie 1995; Jellinek and Card 1990), user preference, comfort, and cost. Other important engineering parameters include sampling rate, resolution, accuracy, and linearity (MacKenzie 1995).

6.3 DIRECT-INPUT DEVICES

A direct-input device has a unified input and display surface. A mouse, by contrast, is an indirect-input device because the user must move the mouse on a surface (a desk) to indicate a point on another surface (the screen). Direct devices such as touch screens, or display tablets operated with a pen, are not necessarily easier to use than indirect devices. Direct devices often lack buttons for state transitions. Occlusion is also a major design challenge: The finger or pen covers the area where the user is pointing, so the user may not be able to see all the options in a pop-up menu, for example. The hand and arm may also occlude visual feedback, dialogs, status indicators, and other controls. Indirect input scales better to large interaction surfaces, since it requires less body movement and also allows interaction at a distance from the display.

Since direct input represents a major trend in both research and industry, we give a detailed treatment here of a number of design dimensions and considerations for direct devices with a more brief assay on indirect devices in Section 6.4 that follows.

In addition to having unified input and display surfaces, with direct-input devices a system's feedback for user input is typically localized to the physical points of contact. Some direct-input devices can sense only a bare finger. Others such as resistive touch screens can sense either a plastic stylus or a bare finger but cannot distinguish one type of contact from the other. Transparent electromagnetic digitizers, such as those found on tablet personal computers, require the use of a special pen and cannot sense touch unless the system integrates a second touch-sensitive digitizer. Some commercially available digitizers based on capacitive coupling can sense a special pen and simultaneously differentiate it from multitouch inputs (Engelhard 2008).

Since there is not yet any "perfect" pen, touch, or combined pen + touch technology, a few of the key questions to ask are as follows:

- Can a digitizer sense touch? If so, is it a "soft-touch" screen or is a definite contact pressure required to sense touch? If significant pressure is required, it makes touch-based dragging operations more fatiguing and more difficult for the user to perform successfully (e.g., without the finger skipping). On the flip side many "hard-touch" screens can be used while the operator wears gloves, making such screens well suited to demanding working environments, whereas capacitive touch screens require contact from bare fingers in order that touch is sensed.
- How many points of contact can a digitizer sense? Many current devices are limited to two to four touch points, which limits the multitouch or hand palm contact signals that a device can reliably sense. In some cases, two-point devices have further limitations (such as line-of-sight issues with optical sensing techniques or reporting only the bounding box of the touch points, rather than the actual touch points themselves, with some touch-screen technologies). On some devices, large-contact-area touches, such as those from a thumb or palm, may be reported as two or more points of contact, which introduces further complexities in the software and interaction design. Likewise, closely spaced contacts may merge into a single contact on many touch screens producing further potential difficulties and technical limitations. Such complexities often result from the use of sophisticated filtering techniques, firmware, and drivers rather than the hardware per se, but nonetheless they pervade commercially available touch devices.
- Can the touch contact area or pen pressure be sensed and reported to applications, or is each contact reduced to a single (x, y) point? If contact area is sensed can applications access the actual image or contour of the contact, or is it only reported as a bounding ellipse (or other simplified representation) of the contact?
- If a stylus is supported at all is a special pen required, or can contact from any hard object be sensed? If a special stylus is required, keep in mind that it is not a matter of *if* but rather of *when* the user loses the stylus. Most special pens on the market are powered by inductive coupling, but some are active pens that require batteries.
- Can pen contacts be distinguished from touch contacts? Capacitive styli are available for many touch screens, although typically these inputs cannot be differentiated from ordinary finger contacts.
- Can pen contacts also be sensed while one is touching the device? Can touch still be sensed while the pen remains in the proximity of the screen? Many current devices stop reporting touch when the pen is in the range of the screen so as to reduce false inputs triggered by palm contact. Also, note that most digitizers can sense only a single pen at a time, and typically one pen cannot be distinguished from another (e.g., although some high-end Wacom display tablets have a "PenID" feature that can distinguish various types of pens in order to support facile swapping between custom brush settings, e.g.).

Standard data sheets for digitizers often leave these questions unanswered, and it may be unclear what brand or type of digitizer an integrated product uses, which makes it difficult to ascertain exactly what a particular device offers. Buyers beware, and ask a lot of questions. On the flip side, if you are designing an application without direct knowledge of the underlying hardware that a user will actually be using to experience your software, then you must somehow choose a design path that takes into account all these possible hardware manifestations. This is akin to successfully guiding a single thread through a gauntlet of needles in the dark. Welcome to the breadth–depth dichotomy.

We stated in Section 6.1 that every input modality is best for something and worst for something else. Ultimately it is the designer's job to know what to use when, for whom, for what, and why. From a technology standpoint, much of this is based on a nuanced understanding of the properties of an input modality. To offer insight into the main issues for commonplace devices, Table 6.1 summarizes the interaction properties shared by pen and touch. We do not characterize these properties as pros and cons to accentuate our belief that almost any property of a device when used appropriately can be advantageous in a design. A number of recent systems that explore combined pen and touch input epitomize this approach to design (Hinckley, Pahud, and Buxton 2010; Hinckley et al. 2010; Zeleznik, Bragdon, Ko et al. 2010;

TABLE 6.1
Comparison of Several Key Properties Shared by Touch and Pen Input Devices

Property	Pen	Touch
Contacts	1 point A single well-defined point	1–10+ contact regions Often with shape information (Cao et al. 2008)
Occlusion	Small (pen tip) But hand still occludes screen	Moderate (fat finger) to large (pinch, palm, whole-hand gestures)
Precision	High Tripod grip/lever arm affords precision, writing, and sketching tasks.	Moderate Nominal target size for rapid acquisition via touch is about 10–18 mm^2. (Vogel and Baudisch 2007) (Sears 1993) (Lewis, Potosnak, and Magyar 1997)
Hand	Preferred hand	Either hand/both hands
Elementary inputs	Tap, drag, draw path	Tap, hold, drag finger, pinch
Intermediary	Mechanical intermediary Takes time to unsheathe the pen. Pen can be forgotten.	None: bare-handed input Nothing to unsheathe, nothing to lose. No lever arm
Acquisition time	High (first use: unsheathe the pen) Moderate on subsequent uses: pen tucked between fingers	Low No mechanical intermediary to acquire
Buttons	Barrel button, eraser (some pens)	None
Activation	Nonzero	Zero (capacitive touch)
force	Tip switch or minimum pressure	Note that resistive touch requires some force.
False inputs	Palm rejection: palm triggers accidental inputs, fingers drag on screen while writing, etc. This is a difficult problem. Designs must accommodate incidental palm contact when it inevitably occurs.	"Midas touch problem" Fingers brush screen, finger accidentally rests on screen while holding the device, etc. "Chess player's syndrome" Device senses touch when none occurs. Common problem on optical touch screens.

Source: Hinckley, K., M. Pahud et al., Paper read at Society for Information Display 2010 Digest, 2010; Hinckley, K., K. Yatani et al., Paper read at 2010 Symposium on User Interface Software and Technology, New York, 2010. With permission.

Brandl et al. 2008; Brandl et al. 2009; Frisch, Heydekorn, and Dachselt 2009). Beyond pen and touch, we believe this is a valuable perspective for a broad class of input devices, modalities, and form factors, as exemplified by other research studies on multimodal input (Bolt 1980; Cohen et al. 1997; Tse et al. 2008; Levine and Ehrlich 1991).

This limited survey shows that pen and touch, although sharing common ground as direct-input modalities, exhibit many important differences, and these again differ substantially from the properties of indirect pointing devices such as the mouse. However, also keep in mind that Table 6.1 is just a summary of some common properties of these devices and the properties of specific pen or touch devices can vary widely, as we now discuss.

Single touch versus multiple touch: Devices capable of detecting touch are often classified colloquially as either touch or multitouch devices, but there is subtle distinction between the two. Single-touch devices, such as traditional resistive touch screens, are adequate for emulating a mouse and for detecting most of the gestures employed in commercial multitouch software today (Potter, Weldon, and Shneiderman 1988).

Multitouch devices can be further classified by the number of finger contacts they are able to detect and track. Multiple contacts are required for a user to perform true multitouch gestures, such as pinch-to-zoom or multifingered grabbing (Krueger, Gionfriddo, and Hinrichsen 1985; Moscovich and Hughes 2006). Still more contacts must be tracked to enable multiple users to perform multitouch gestures at the same time, as desired for tabletop interfaces, for example.

Many devices reputed to be multitouch can, in fact, only sense limited information about touch contacts, such as the bounding box (rather than the actual *x*, *y* coordinates) of the touches. This class of devices includes the DiamondTouch table (Dietz and Leigh 2001) and some optical touch-sensing technologies. These are sometimes referred to as "1.5D" devices since they do not support two fully independent degrees of freedom for each touch point. Nonetheless, such devices can implement a wide array of multitouch interaction techniques (Wu and Balakrishnan 2003; Forlines and Shen 2005).

Pressure and contact area sensing: Pressure is the measure of force that a user exerts on an input device.

Pressure sensing is often confused with contact area sensing. True pressure sensing is supported by many pen-operated devices, but typically only contact area can be sensed by touch-screen interfaces. The two are related; although contact area is a useful proxy of pressure, it cannot be treated as a true equivalent of pressure for the purposes of interface and interaction technique design. For example, a woman with long fingernails who touches a screen will produce a touch with a large contact area, even though the actual force applied may be very light. With contact area, rather than relying on an absolute degree of contact one should emphasize on *changes in contact area* as a more controllable parameter that users can modulate, such as by rolling one's finger in contact with the display (Benko, Wilson, and Baudisch 2006). Contact area has been demonstrated to add states to touch devices (Forlines and Shen 2005), allowing a type of hover state when touching lightly and then committing to a particular action when the user presses more forcefully. Used as a continuous parameter, contact area can enable a continuous control of the thickness of a paint stroke or set the *z*-order of on-screen objects (Davidson and Han 2008), for example.

However, some touch devices do offer true pressure-sensing capability. For example, some laptop touchpads include sensors that measure the force exerted on the pad. In multitouch systems, such approaches typically provide only an aggregate of forces across all contact areas rather than an independent measure of pressure for each touch.

Users are able to perceive up to seven distinct levels of pressure, provided there is suitable visual feedback (Ramos, Boulos, and Balakrishnan 2004). Pressure can then be used for a variety of interaction techniques such as mode switching (Agarawala and Balakrishnan 2006) or continuous adjustment of parameters while drawing a pen stroke (Ramos and Balakrishnan 2005; Ramos, Boulos, and Balakrishnan 2004; Ramos and Balakrishnan 2006).

Hand postures and shape-based input: Devices that can sense not only points of contact but also shape of the contact region can allow for richer input (Krueger 1991; Cao et al. 2008). Researchers have also used the sensed contact region to compute a user's perceived point of contact for high-precision touch interaction. For example, Holz and Baudisch (2010) used a fingerprint sensor to determine a finger's contact, orientation, and position and mapped these parameters to more precisely determine a user's intended touch location. Subtler uses of shape-based input have demonstrated the advantage of using a geometric representation of the contact area instead of the traditional reduction of all touch information to a single point, so as to enable rapid interaction with multiple widgets (Moscovich 2009) or to support greater expressiveness from each touch (Cao et al. 2008; Zeleznik, Bragdon, Adeputra et al. 2010). A challenge for the effective and practical use of shape-based touch interaction is the lack of a commonly accepted input event and interaction model for software development and user interface design. It remains difficult to develop practical multitouch applications that wander too far from the path followed by the traditional point-based hit testing model of interaction. Probabilistic approaches offer one potential solution to this dilemma (Schwarz et al. 2010).

In-air hand postures and direct sensing beyond the display: Recent approaches have also demonstrated the utility of in-air hand postures as opposed to those that occur while in contact with a device (Grossman, Wigdor, and Balakrishnan 2004; Hilliges et al. 2009; Wilson and Benko 2010). For example, the SecondLight system can see through the display and can project both light and sense interactions in the volume above the display itself. Using this system, Hilliges (Hilliges et al. 2009) used cameras to detect simple postures both in air and when in contact with the display, using the moment of contact as a state transition to delimit interactions. This enables both the user and the system to agree on, and differentiate between, hand gestures that occur in contact with the display as opposed to incidental movements of hands in the air. Thus, detecting and differentiating system response based on physical contact with the screen delivers an experience with discrete and controllable states, even though the device perceives similar hand movements in either case. Wilson and Benko (2010) further explore the possibilities of this space using multiple depth cameras and projectors.

Furthermore, this demonstrates once again how direct input is about much more than touch alone: The positions of users relative to the display (Ballendat, Marquardt, and Greenberg 2010), motions of their hands above the display (Tang et al. 2010), and posture of the hand as it comes into contact with the display (Holz and Baudisch 2010) are all forms of direct input that can extend and enrich user interfaces. What is critical to note about these examples is that there is no attempt to use in-air gesturing to replace the mouse, simulate text entry on a virtual keyboard, or replace established touch gestures. Rather, the aforementioned examples leverage the distinct properties of spatial sensing to provide new capabilities that are differentiated well from direct touch and other inputs.

Finger differentiation, user differentiation, and user identification: Some touch devices are able to determine which of a user's fingers are in contact with the display. For example, the Microsoft Surface images enough of the hand in proximity to the display that it is often possible to determine which fingers of a hand are in contact with the display (Lepinski, Grossman, and Fitzmaurice 2010). Some systems can identify the source of contacts, such as those using capacitive coupling techniques (Dietz and Leigh 2001), so that a touch from one user can be distinguished from the touch of another user. User identification is a stronger form of differentiation that persists in the distinction between multiple users across sessions. One way to achieve this is through the sensing and identification of a user's fingerprints while he or she is touching a display (Holz and Baudisch 2010), although current demonstration systems cannot achieve such sensing in real time. Computer vision technologies have also been demonstrated, which can identify users or distinguish which fingers of a user's hand are touching the display, based on visual features.

Parallax: Parallax error is a mismatch between sensed input position and apparent input position due to viewing angle. *Display parallax* is the displacement between the sensing and display surfaces. Of course zero parallax is ideal, but a rule of thumb used by industry practitioners is that if the display parallax is less than about 2 mm then its practical impact is not that significant. *Transducer parallax* is caused by any additional parallax error that may result from the offset between the tip of a mechanical intermediary and the actual component that is sensed, such as the coil that sits higher up in the body of Wacom electromagnetic pens. If tilt angles of the pen are known, this offset can be corrected.

Latency: Latency is a problem for all interactive systems, but its effects are particularly insidious for direct-input devices because any discrepancy between the actual position of fingers or the pen and the position currently sensed by a system becomes immediately obvious to the user. Latency is the end-to-end measure of the time elapsed between the moment a physical action is performed by a user and the moment the system responds to it with feedback that the user can perceive. This round-trip time between cause and effect comes from many sources; it is difficult to minimize and impossible to eliminate in practical systems design. Sources of latency include hardware sampling rate, time a system takes to report samples to the operating system as well as report events to applications, time required by software to compute results, time required to refresh the frame buffer, and the physical screen's rate of update to make the results visible to the user. Seow (2008) offers a detailed study on the effects of latency on user perception. In experiments, latency typically exhibits strong negative effects on user performance starting at about 100 milliseconds (Card, Mackinlay, and Robertson 1991; MacKenzie and Ware 1993). In modern direct-input systems with pen or touch input, latencies far less than 100 milliseconds must be achieved (Card, Mackinlay, and Robertson 1991; MacKenzie and Ware 1993) if one wishes users to perceive the interface as crisp and responsive.

6.4 INDIRECT-INPUT DEVICES

An indirect-input device is one that does not provide input in the same physical space as the output. Indirect devices eliminate occlusion of the screen by the user's hand and fingers. However, typically they require more explicit feedback and representation of the input device (such as a cursor), intended target on the screen (such as highlighting icons when the cursor hovers over them), and current state of the device (such as whether a button is held or not).

Virtual devices: Most operating systems treat all input devices as virtual devices, which tempts one to believe that input devices are interchangeable; however, details of what the input device senses, how the device is held, the presence or absence of buttons, and many other properties can significantly impact the interaction techniques—and hence, the end-user tasks—that a device can effectively support. Although the virtual devices abstraction has been a successful strategy in addressing the breadth–depth dichotomy for indirect devices, largely because the tracking symbol serves as an intermediary between the type of device and the interaction techniques that it supports, it is becoming increasingly clear that the virtual devices abstraction is far less satisfactory when extended to direct-input devices.

Indirect digitizing tablets and absolute versus relative input: An absolute input device senses the position of an input and passes this message along to the operating system. Relative devices sense only changes in position. Digitizing tablets can operate either in absolute mode, with a fixed control-to-display (C:D) gain between the tablet surface and the display, or in relative mode, in which the tablet responds only to the motion of a stylus or finger. If the user touches the stylus or finger to the tablet in the relative mode, the cursor resumes motion from its previous position; in absolute mode, it jumps to the new position. Absolute mode is generally preferable for tasks such as drawing, handwriting, tracing, or digitizing, but relative mode may be preferable for traditional desktop graphical user interaction tasks such as selecting icons or navigating through menus. Digitizing tablets thus allow coverage of many tasks (Buxton, Hill, and Rowley 1985), whereas mice can operate only in relative mode.

Mixing indirect and direct inputs: It is possible to emulate the properties of an indirect-input device using a direct one, simply by offsetting the device's apparent input target from its physical location. This leads to an interesting class of hybrid techniques that mix direct and indirect representations within the user experience of a direct-input device (Buxton, Hill, and Rowley 1985; Brown, Buxton, and Murtagh 1990; Sears and Shneiderman 1991; McCallum and Irani 2009). For example, HybridPointing provides a cursor that remains directly beneath an input stylus, but can also decouple to allow pointing at distant targets on very large displays without great physical movement (Forlines, Vogel, and Balakrishnan 2006). Such techniques also support high precision pointing on touch screens (Vogel and Baudisch 2007).

6.4.1 Brief Tour of Indirect-Input Devices

The following tour discusses important properties of several common indirect-input devices, such as mice, joysticks, touchpads, trackballs, and stand-alone touch tablets and pen digitizer tablets:

Touchpads: Touchpads are small, touch-sensitive tablets often found on laptop computers. Touchpads usually use relative mode for cursor control because they are too small to map to an entire screen, but most touchpads also have an absolute mode to allow interactions such as character entry or sliding along the edge of the pad to scroll. The small size of touchpads necessitates frequent clutching, and touchpads can be awkward to use while holding down a button unless the user employs his or her other hand. Traditionally, touchpads have supported clicking by recognizing tapping or double-tapping gestures, but accidental contact (or loss of contact) can erroneously trigger such gestures (MacKenzie and Oniszczak 1998). To reduce these types of problems, some modern touchpads often include a microswitch underneath the pad, so that pressing down the pad produces a mechanical click.

Multitouch pads: Modern touchpads are increasingly becoming multitouch devices as opposed to the single-touch models that were commonplace in the past. The multitouch interaction model for indirect touchpad typically differs from that for direct-touch displays. If both the devices are natural multitouch devices, then why are two different interaction models necessary? The indirect touchpad must support relative cursor control and typically single touch is mapped to move the cursor. Thus two fingers are required to scroll or pan documents. With a direct-touch input device, cursor tracking is not necessary and hence single touch can pan and scroll. This underscores why characterizing a device as simply a touch device is not sufficient to understand its properties. Furthermore, it underscores why the assumption that touch inherently produces a natural interface is a fallacy. Finally, this demonstrates why an inconsistency between two related devices—here, an indirect- versus a direct-touch device—is often necessary to produce a user experience that feels logically consistent to the user given the device and form factor at hand.

Indirect multitouch pads can also support additional novel techniques such as targeting many degrees of freedom to a single cursor position or representing each point of contact with a separate cursor (Moscovich and Hughes 2006). Such models have also been explored in the context of mice augmented with multitouch input surfaces (Villar et al. 2009; Benko et al. 2010). An alternative to cursors that can be useful on both direct and indirect devices is to present a "shadow" of the whole hand (Krueger 1991; Wigdor, Forlines et al. 2007a; Tang et al. 2010).

Mice: Douglas Englebart and colleagues (English, Englebart, and Berman 1967) invented the mouse at the Stanford Research Institute, Menlo Park, California. More than 40 years later, use of the mouse persists because its properties provide a good match between human performance and the demands of graphical interfaces (Balakrishnan et al. 1997). For typical pointing tasks on a desktop computer, one can point with the mouse about as well as with the hand (Card, English, and Burr 1978). Because the mouse stays put when the user releases it (unlike a stylus, e.g.), the user can reacquire it quickly and designers can integrate multiple buttons or other controls on its surface. Users exert force on mouse buttons in a direction orthogonal to the mouse's plane of motion, thereby minimizing inadvertent motion. Finally, with mice all the muscle groups of the hand, wrist, arm, and shoulder of users contribute to pointing, allowing high performance for both rapid, coarse movements and slow, precise movements (Guiard 1987; Zhai, Buxton, and Milgram 1996). These advantages suggest the mouse is hard to beat; it will remain the pointing device of choice for desktop graphical interfaces for many more years to come.

Trackball: A trackball senses the relative motion of a partially exposed ball in two degrees of freedom. Trackballs have a small working space (footprint), afford use on angled surfaces, and sometimes are weighted to afford spinning the ball with physical inertia. Trackballs may require frequent clutching movements because users must lift and reposition their hand after rolling the ball through a short distance. The buttons are located to the side of the ball, which can make them awkward to hold while rolling the ball (MacKenzie, Sellen, and Buxton 1991). A trackball engages different muscle groups compared to a mouse, offering an alternative for users who experience discomfort when using a mouse. Trackballs are also well suited to non-preferred-hand input in combination with a mouse (Bier et al. 1993).

Isometric joysticks: An isometric joystick (e.g., the IBM Trackpoint) is a force-sensing joystick that returns to center when released. Most isometric joysticks are stiff, offering little feedback of the joystick's displacement. The rate of cursor movement is proportional to the force exerted on the stick; as a result, users must practice in order to achieve good cursor control. Isometric joysticks are particularly appealing when space is at a premium (Douglas and Mithal 1994; Rutledge and Selker 1990; Zhai, Smith, and Selker 1997).

Isotonic joysticks: Isotonic joysticks sense the angle of deflection. Some hybrid designs blur the distinctions between isometric and isotonic joysticks, but the main questions that characterize the design space are the following:

- Does the joystick sense force or angular deflection?
- Does the stick return to center when released?
- Does the stick move from the starting position?

For a more thorough discussion on the complex design space of joysticks, we recommend the study by Lipscomb and Pique (1993).

6.4.2 Summary of and Perspective on Indirect-Input Devices

Although one may feel that many of the indirect-input devices are antiquated in the context of modern system design trends, such devices still have their uses and can offer important lessons to a student of HCI. Furthermore, often consoles or workstations for dedicated and highly specialized tasks, such as driving a car, still incorporate many creative combinations of such devices in conjunction with physical buttons, switches, and dials. To anyone who has attempted the dangerous exercise of operating a touch-screen navigation system while driving, it should come as no surprise that rich spatial arrangements of physical input devices with complementary and specialized roles offer a highly effectively solution while placing fewer demands on an operator's visual attention (e.g., Fitzmaurice, Ishii, and Buxton 1995). Hence, combinations of indirect-input devices are ideally suited for many applications and remain important to understand for interaction designers worthy of their name.

6.5 INPUT DEVICE STATES

Any device that returns a coordinate can be considered a pointing device; but there remains a critical missing ingredient: the *event* that a device generates, such as when a finger comes into contact with a screen, a button is depressed on a mouse, or a pen leaves the proximity of a digitizer. These events trigger state transitions, which in turn are the building blocks of interaction techniques. All of this seems blatantly obvious until one realizes that there are subtle discrepancies between the events and properties sensed by various devices while in different states. This innocent façade of greenery conceals a nightmarish thicket that can entangle the designer in stinging nettles and poisonous ivies of failed or severely compromised interaction techniques.

There is a fundamental mismatch between the demands of traditional graphical interfaces and the states and events that can be sensed by devices such as touch screens, touchpads, and pen-operated devices, which makes it difficult for such devices to support the full set of desktop GUI primitives, including click, drag, double-click, and right-click. There is no easy solution that does not involve design compromises. When considering such devices, to make device limitations and differences concrete, diagramming all the states and transitions can be an enlightening exercise.

Input devices in general support three possible states (Table 6.2): (1) out-of-range, (2) tracking, and (3) dragging states. Practitioners refer to these as state 0, state 1, and state 2, respectively, of the three-state model (Buxton 1990b). This model is useful for understanding the relationship between events sensed by an input device and demands of interaction techniques.

The three-state model describes the mouse as a two-state device supporting state 1, the cursor-tracking state, as well as state 2, the dragging state (Figure 6.3). State 1 provides cursor feedback of the screen position on which the device

TABLE 6.2
Comparison of Several Key Properties Shared by Touch and Pen Input Devices

State	Description
0	Out of range: The device is not in its physical tracking range.
1	Tracking: Device motion moves only the cursor.
2	Dragging: Device motion moves objects on the screen.

Source: Hinckley, K., M. Pahud et al., Paper read at Society for Information Display 2010 Digest, 2010; Hinckley, K., K. Yatani et al., Paper read at 2010 Symposium on User Interface Software and Technology, New York, 2010. With permission.

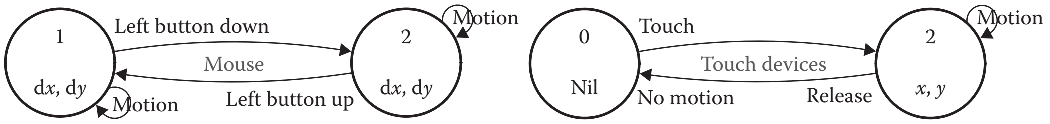


FIGURE 6.3 State-transition diagram for standard mouse and touch input devices. Each device senses two states, but critically not the same two states.

will act, whereas state 2 allows the user to drag an object by holding down the primary mouse button while moving the device. The mouse senses movement in both the tracking and dragging states, as represented by the (d*x*, d*y*) in each state in Figure 6.3 (left), which is shorthand to indicate relative motion tracking capability (Hinckley, Czerwinski, and Sinclair 1998).

Many touch-activated devices, such as touch screens and touchpads, are also two-state devices, but *they do not sense the same two states as the mouse* (Figure 6.3, right). For example, a mobile phone touch screen can sense a finger when it is in contact with the display; this is the equivalent of the mouse's dragging state (state 2). The touch screen can also sense when the finger is removed from the screen, but once the finger breaks contact this enters state 0 (out-of-range state), where no motion can be detected (emphasized by the annotation Nil in state 0 of Figure 6.3 [right]). Thus, although the mouse and the touch screen both sense two states, the lack of a second motion-sensing state on the touch screen means that it will be difficult to support the same interaction techniques as that of the mouse. For example, should sliding one's finger on the device move a cursor or drag an object? The designer must choose one; a touch-screen or touchpad device cannot support both behaviors at the same time.

If we add pens for tablet computers to the mix, we now have a third state: Many pens can sense when they enter or leave the proximity of the screen (Figure 6.4). But despite the addition of the third state, the input events sensed by the pen still differ from those sensed by the mouse, as well as those sensed by touch. Hence, all three devices are mutually incompatible in subtle ways and interactive techniques that attempt to support all these devices at the same time will likely encounter various inconsistencies and usability problems as a result.

Pen and touch devices often lack a button suitable for right-click, leading to the heavy use of awkward interface Band-Aids such as a touch-and-hold solution to invoke context menus. Even if a pen does include a barrel button, it is relatively slow to access as well as being prone to inadvertent activation (Li et al. 2005). The press-and-hold solution has become standard on mobile devices, but the time-out introduces an unavoidable delay. For rapid activation the time-out should be as short as possible, whereas to avoid inadvertent activation paradoxically the time-out must be as long as possible. A 500-millisecond time-out offers a reasonable compromise, but as most commercial devices assume that context menu invocation occurs only infrequently they use delays of about 1000 milliseconds to further tip the design balance toward reducing accidental activations. Techniques designed for pen-operated devices (Apitz and Guimbretière 2004; Kurtenbach and Buxton 1991a; Kurtenbach and Buxton 1991b; Moran, Chiu, and van Melle 1997) should afford rapid and unambiguous activation of one of several possible actions as fundamental building blocks (Hinckley, Baudisch et al. 2005) to avoid inefficient or highly modal interactions (Hinckley et al. 2006).

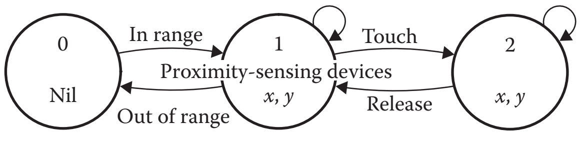


FIGURE 6.4 State-transition diagram for proximity-sensing devices, such as a pen on a tablet personal computer.

Similar issues plague other interaction modalities also, such as motion-sensing mobile devices (Hinckley, Pierce et al. 2005; Hinckley and Song 2010), camera-based tracking of the hands (Wilson and Oliver 2003), and spatial input devices (Hinckley et al. 1994). All these techniques require a method for users to move a device or their hands without accidentally performing an action, which speaks to the appeal of hybrid solutions such as the combination of spatial sensing with direct-touch input (Wilson and Benko 2010). State transitions thus form fundamental indications of intent that are essential to the construction of rapid, dependable, and habit-forming interaction techniques.

6.6 COMPOSITION OF USER TASKS

One way of reasoning about input devices and interaction techniques is to view a device or technique in light of the tasks that it can express. But what sort of tasks are there?

6.6.1 Elemental Tasks

Although computers can support many activities, at the input level some subtasks appear repeatedly in GUIs, such as pointing to a target on the screen or typing a character. Foley, Wallace, and Chan (1984) identified "elemental" tasks including text (entering symbolic data), select (indicating objects from a set of alternatives), position (pointing to a screen coordinate), and quantify (specifying an exact numeric value). However, if these are elemental tasks then where do devices such as global positioning system readers, cameras, and fingerprint scanners fit in? These offer new elemental data types of location, images, and identity, respectively. Elemental tasks are difficult to fully enumerate, because advances in technology continue to yield data types that enable new tasks and scenarios of use. Furthermore, the perspective of what tasks are elemental depends in part on the input devices, as well as the interaction design, through which an interface expresses them.

6.6.2 Compound Tasks and Chunking

A fundamental problem with the elemental task approach is that the level of analysis for elemental tasks is not well defined. For example, a mouse indicates an (x, y) position on the screen, but an Etch A Sketch separates positioning into two subtasks by providing a single knob for x and a single knob for y (Buxton 1986b). If position is an elemental task, why must we subdivide this task for some devices but not others? One way to resolve this puzzle is to view all tasks as hierarchies of subtasks (Figure 6.5). Whether or not a task is

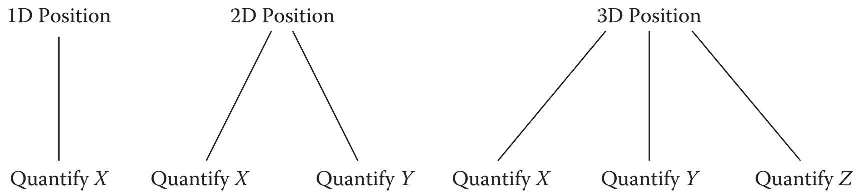


FIGURE 6.5 User tasks conceived as unitary actions, or as assemblies of subtasks.

elemental depends on the input device being used: The Etch A Sketch supports separate *quantify X* and *quantify Y* tasks, whereas the mouse supports a compound *2D position* task (Buxton 1986). Now, if a user wishes to indicate a point on the screen, the integrated (*x*, *y*) position of the mouse offers the best solution. But if the user wishes to precisely specify an *x* coordinate independent of the *y* coordinate, then the Etch A Sketch more directly expresses this concept.

From the user's perspective, a series of elemental tasks may seem like a single task. For example, scrolling a web page to click on a link could be conceived as an elemental 1D positioning task followed by a 2D selection task or it can be viewed as a compound navigation/selection task (Buxton and Myers 1986b). An interaction technique can encourage the user to work at the higher level of the compound task, for example, by scrolling with one hand while pointing to the link with the other hand. This is known as "chunking."

These examples show that the choice of device influences the level at which a user is required to think about the individual actions that must be performed to achieve a goal. The design of input devices and interaction techniques can help to structure the interface in such a way that there is a more direct match between user's tasks and the low-level syntax of individual actions that must be performed to achieve those tasks. The choice of device and technique thus directly influences the steps required of the user and hence the apparent complexity of an interface design (Buxton 1986).

6.6.3 Phrasing

Most of the examples of chunking mentioned in Section 6.6.2 revolve around the integration of multiple degrees of freedom to support higher-level tasks. Another implement in the toolbox of the interface designer to afford the design of compound tasks is *phrasing*, which is the use of muscular tension to maintain a temporary state that "glues together" multiple subtasks (Buxton 1995). A good example is that of a pull-down menu for which holding down the mouse button integrates the tasks of activating the menu, sliding the pointer to the desired menu item, and then lifting the button to select the menu item. The muscular tension from maintaining contact with a touch screen or from holding down a button on a mouse provides continuous and salient proprioceptive feedback to the user that the system is in a temporary state, or mode, where movement of the input device navigates the menu instead of moving the cursor. Another key property of phrasing is that closure is inherent in the means used to introduce the phrase: Simply releasing the mouse button returns the system to its default state. Obviously, this approach has its limits as the user cannot maintain muscular tension for very long without discomfort and fatigue, but it has been shown to reduce mode errors for frequent and temporary mode switches (Sellen, Kurtenbach, and Buxton 1992). Although the preceding example of the pull-down menu requires a mouse button, phrasing can also be achieved on direct-input devices by maintaining finger contact with a touch screen (Hinckley et al. 2010) or by holding a pen against a digitizer while articulating a compound gesture (Kurtenbach and Buxton 1991; Apitz and Guimbretière 2004).

6.7 EVALUATION AND ANALYSIS OF INPUT DEVICES

Beyond standard usability engineering techniques, there are several techniques tailored to the study of input devices and interaction techniques. Representative tasks (Buxton 1995), such as target acquisition, pursuit tracking, freehand drawing, and dragging versus tracking performance (MacKenzie, Sellen, and Buxton 1991), can be used to formally or informally evaluate devices. Here, we focus on formal analysis using Fitts's law, the steering law, and the keystroke-level model (KLM), but we begin with a high-level perspective on how to honestly evaluate interaction techniques in general.

6.7.1 Evaluating the True Cost of a Technique in the Context of Surrounding Operations

When evaluating an interaction technique, one must fully consider the true cost to invoke a command (Dillon, Eday, and Tombaugh 1990) as well as the context of surrounding actions that are necessary to set up and recover from a particular action (Appert, Beaudouin-Lafon, and Mackay 2004; Hinckley et al. 2006). Commands that act on a selection are one example: The interaction technique used to select objects can influence how effectively one can act on the selection (and vice versa). Indeed, if one views selection–action phrases as a compound interaction technique rather than studying selection in isolation from command activation, new approaches that optimize a compound task may become possible (Kurtenbach and Buxton 1991; Hinckley, Baudisch et al. 2005; Apitz et al. 2004). A discussion of which techniques are most appropriate for a given task can be found in the work by Mackay (2002).

As another "toy" example, let us say that a user's task is to create a single square. What is the best interaction technique to perform this task? Tapping on a mechanical button is very fast. Perhaps the user could tap a button on a pointing device to create a square centered on the current cursor location. But what if the user must be able to produce circles, triangles, and polygons also? Creating a square from just a button tap will no longer work because one needs a way to specify the type of shape to be created. Perhaps depressing a button to invoke a menu where one can stroke in a particular direction to create the desired shape will work well (Kurtenbach and Buxton 1991). What this toy example makes clear is that one cannot optimize the *create square* interaction in isolation. One must also consider the set of surrounding interactions and tasks that together comprise the user interface.

We must consider this same perspective when evaluating interaction techniques. To extend the aforementioned example a bit further, what if the user also wishes to be able to make notations in free hand with the same input device? One then has to manage the mode of the device, with a *command mode* to create the shapes and an *inking mode* to make freehand strokes. The cost of invoking the create square command now depends on the total time taken to enter the mode, articulate the create square command, and then return back to the original mode. We cannot just measure the time it takes to draw the create square gesture. Even costs such as repositioning the pointer to the working area of the user interface or the user's mental preparation time to plan the next step must be considered. We cannot honestly evaluate the *true cost* of articulating the command without a full and diligent account of its entire lifecycle.

Furthermore, the context of surrounding operations can influence the efficacy of a technique as well. What if the user's task was to create many squares in a row? What if it was to interleave the creation of single squares with the drawing of freehand marks? Our view of which technique is most effective in a given context may depend on such usage patterns (Hinckley et al. 2006; Appert, Beaudouin-Lafon, and Mackay 2004; Guimbretiere, Martin, and Winograd 2005).

6.7.2 Fitts' Law

Fitts' law has been so widely applied to the design and evaluation of input devices that the mere mention of it is taken as a dirty word in some quarters. Nonetheless, it is a highly sensitive tool for evaluating input devices and techniques when used appropriately.

Fitts' law (Fitts 1954) is an experimental paradigm that has been widely applied to the comparison and optimization of pointing devices. Fitts' law is used to measure how effectively a pointing device can acquire targets on a screen. This law was first applied to the study of input devices by Card, English, and Burr (1978); it is now a standard for device comparisons (Douglas, Kirkpatrick, and MacKenzie 1999). Fitts' law can be applied to remarkably diverse task conditions, including rate-controlled devices (MacKenzie 1992a), area cursors (Kabbash and Buxton 1995), scrolling (Hinckley et al. 2002), and zooming (Guiard et al. 2001). For further guidance on conducting studies on Fitts' law, see the studies by Douglas, Kirkpatrick, and MacKenzie (1999); MacKenzie (1992); and Raskin (2000).

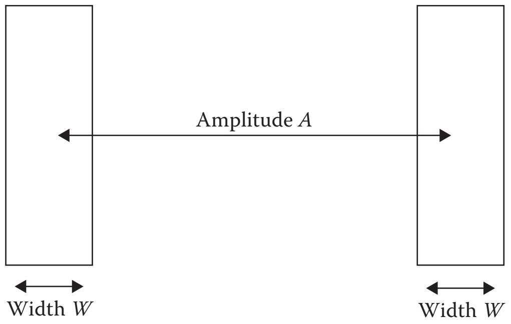


FIGURE 6.6 Canonical Fitts' task, with a pointing amplitude A between targets of width W.

The standard Fitts' task paradigm measures the movement time MT between two targets separated by amplitude A, with a width W of error tolerance (Figure 6.6). Fitts' law states that a logarithmic function of the ratio of A to W predicts the average movement time MT. The Fitts' law formulation typically used for input device studies is as follows:

$$MT = a/b \log_2\left(A/W + 1\right) \tag{6.1}$$

Here, the constants a and b are coefficients that fit to the average of all observed MT for each combination of A and W in the experiment. One calculates a and b via linear regression using a statistical package or spreadsheet. The constants a and b depend heavily on the exact task setting and input device, so be wary of substituting "typical" values for these constants or of comparing constants derived from different studies.

Psychomotor interpretations of Fitts' law have been proposed (Douglas and Mithal 1997). However, since the law does not characterize individual pointing movements but rather the central tendency of a large number of pointing movements, the law may simply reflect information-theoretic entropy (MacKenzie 1989).

6.7.3 Steering Law and Minimum Jerk Law

Steering a cursor through a narrow tunnel, as required to navigate a pull-down menu, is not a Fitts' task because steering requires a continuous accuracy constraint: The cursor must stay within the tunnel at all times. For a straight-line tunnel (Figure 6.7) of width W and length A, for example, the steering law models movement time as a linear function of A and W:

$$MT = a/bA/W \tag{6.2}$$

The steering law can alternatively model instantaneous velocity (Accot and Zhai 1997). A limitation of the law is that it models only the successful completion of a task; errors are not considered.

The minimum jerk law (Viviani and Flash 1995) characterizes the dynamics of motions that lack a continuous accuracy constraint. No one has yet formulated a universal law that handles varying accuracy constraints and curvature (Lank and Saund 2005).

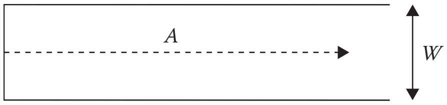


FIGURE 6.7 Steering through a tunnel of length *A* and width *W*.

6.7.4 Keystroke-Level Model and the Goals, Objects, Methods, and Selection Rules Analysis

The KLM is an engineering and analysis tool that can be used to derive a *rough estimate* of the time needed for expert users to complete a routine task (Card, Moran, and Newell 1980). To apply KLM, count the elemental inputs required to complete a task, including keystrokes, homing times to acquire input devices, pauses for mental preparation, and pointing at targets. For each elemental input, substitute a constant estimate of the average time required using the values from the study by Card, Moran, and Newell (1980) or by collecting empirical data (Hinckley et al. 2006), and sum them to yield an overall time estimate. The model assumes error-free execution, so it cannot estimate time for the problem-solving behaviors of novices; but it does employ several heuristics to model mental pauses (Raskin 2000).

The goals, objects, methods, and selection rules (GOMS) models extend KLM (John and Kieras 1996). Some GOMS models can account for user knowledge and the interleaving of tasks, but they are more difficult to apply than KLM. Both GOMS and KLM models are engineering tools that produce estimates for expert completion time of routine tasks. These models do not replace the need for usability testing and evaluation, but they do offer a means to assess a design without implementing software, training end users, and evaluating their performance (Olson and Olson 1990). Physical articulation times derived from KLM or GOMS analyses can also be used to interpret results of empirical studies (Hinckley et al. 2006).

6.8 TRANSFER FUNCTIONS: HOW TO TRANSFORM AN INPUT SIGNAL

A transfer function is a mathematical transformation that scales the data from an input device. Typically, the goal is to provide more stable and more intuitive control, but one can easily design a poor transfer function that hinders performance. Here we discuss some simple transformations that are commonly used for various interaction techniques, although the reader should be aware that a deep understanding of the topic requires expertise in control theory, which is well beyond the scope of this chapter (and indeed beyond the expertise of its authors):

Appropriate mappings: A transfer function that matches the properties of an input device is known as an appropriate mapping. For force-sensing input devices, the transfer function should be a force-to-velocity function; for example, the force one exerts on an isometric joystick controls the speed at which the cursor moves. Other appropriate mappings include position-to-position and velocity-to-velocity functions, used with tablets and mice, respectively.

A common example of an inappropriate mapping is calculating a velocity based on the position of the mouse cursor, such as automatically scrolling a document when selecting a large region of text that extends beyond a single screen. The resulting input is difficult to control, and this inappropriate rate mapping is only necessary because the operating system clips the cursor to the screen edge. A better solution would be to ignore the cursor position and instead use the relative position information reported by the mouse to directly control the change of position within the document.

Self-centering devices: Rate mappings suit force-sensing devices or other devices that return to center when released (Zhai 1993; Zhai, Smith, and Selker 1997). This property allows a user to stop quickly by releasing the device. The formula for a nonlinear rate mapping is as follows:

$$\mathrm{d}x = Kx^a \tag{6.3}$$

Where x is the input signal, dx is the resulting rate, K is a gain factor, and a is a nonlinear parameter. The best values for K and a depend on the details of a device and application, and appropriate values must be identified by experimentation or optimal search (Zhai and Milgram 1993). Many commercial devices use more complex mappings (Rutledge and Selker 1990).

Motion-sensing devices: Desktop systems use an exponential transformation of mouse velocity, known as an acceleration function, to modify cursor response (Microsoft Corp. 2002). Acceleration functions do not directly improve pointing performance, although they do reduce the footprint required by a device (Jellinek and Card 1990), which may lead to greater comfort or less frequent clutching (Hinckley et al. 2002).

Absolute devices: Absolute devices offer a fixed, 1:1 control-to-display mapping. This is common in touch-screen and pen inputs. Relative transfer functions may also offer a 1:1 mapping of movements.

Mixed-mode absolute/relative mappings: It is possible to temporarily violate the 1:1 control-to-display mapping of absolute devices such as touch screens by damping small motions to provide fine adjustments; large motions revert to an absolute 1:1 mapping (Sears and Shneiderman 1991). A drawback of such techniques is that cursor feedback in the tracking state becomes the default behavior of the device rather than dragging (Buxton 1990); but researchers have demonstrated ways to overcome this problem by only adjusting cursor position beneath the finger, automatically escalating precise pointing interactions to indirect input, or providing means for a user to quickly switch between direct and indirect modes (Benko, Wilson, and Baudisch 2006; Vogel and Baudisch 2007; Forlines, Vogel, and Balakrishnan 2006). Other hybrids of direct and indirect devices, with mixed relative and absolute input mappings, have also been explored.

6.9 FEEDBACK: WHAT HAPPENS IN RESPONSE TO AN INPUT?

From the technology perspective, one can consider feedback as active or passive. Active feedback is under computer control; passive feedback is not and may result from internal sensations within a user's own body, such as muscle tension from holding down a button, or from the physical properties of a device, such as the feel of clicking its buttons.

Good industrial design guides a user to the appropriate affordances for a device as soon as he or she holds it and perhaps even suggests its intended uses before the user even touches it (Norman 1990; Buxton 2007). Mechanical sounds and vibrations produced by a device provide positive feedback for the user's actions (Lewis, Potosnak, and Magyar 1997). The shape of the device and the presence of tactile landmarks can help users acquire and orient a device without having to look at it (Hinckley et al. 1998).

6.9.1 Proprioceptive and Kinesthetic Feedback

Internal sensations of body posture, motion, and muscle tension (Burdea 1996; Gibson 1962) may allow users to feel how they are moving an input device without looking at the device or receiving visual feedback on a display. This is important when the user's attention is divided between multiple tasks and devices (Balakrishnan and Hinckley 1999; Fitzmaurice and Buxton 1997; Mine, Brooks, and Sequin 1997). Muscular tension can help to phrase together multiple related inputs (Buxton 1995) and may make mode transitions more salient to the user (Hinckley et al. 2006; Raskin 2000; Sellen, Kurtenbach, and Buxton 1992; Hinckley et al. 2010).

6.9.2 Kinesthetic Correspondence

With time, users can adapt to any consistent mapping (Cunningham and Welch 1994) so long as the mapping between input and display is a planar reorientation. Despite this ability, graphical feedback on the screen ideally should correspond with the direction in which a user moves an input device (Britton, Lipscomb, and Pique 1978). If the user moves a device to the left, then the object on the screen should likewise move to the left; however, users can easily adapt to certain kinds of noncorrespondences: When the user moves a mouse forward and backward, the cursor actually moves up and down on the screen; if the user drags a scrollbar downward, the text on the screen scrolls upward. Researchers have also found that the dimensions of an input device should match the perceptual structure of a task (Jacob et al. 1994).

6.9.3 To Cursor or Not to Cursor (with Direct Input)

The cursor serves as a proxy for the user when engaged in indirect input. When using direct input, a user's finger or a mechanical intermediary functions as its own indicator for a well-calibrated system (Potter, Weldon, and Shneiderman 1988). Despite this, there are uses for a cursor. In devices that can sense a hover state prior to the finger or stylus coming into contact with a digitizer (Buxton 1990), the cursor serves to indicate to the user which target will be selected before they commit by touching the surface of the device and it serves as a precise indicator of the contact location (Sutherland 1964). Further, iconic cursors serve as useful indicators of state (Tilbrook 1976). Finally, the presence of the cursor and its response to user input gives feedback to the user that the system is active, tracking, and ready to receive commands.

6.9.4 Echo Feedback versus Semantic Feedback

Input devices, through the lens of interaction techniques, process a stream of sensor values to yield a logical result. A choice can be made as to whether a system's feedback echoes unprocessed sensor data back to the user (here is what the system sees; Krueger 1991) or instead provides a semantic representation of the user's state (here is what the system knows) like the cursor (Sutherland 1964). Traditional systems have trended toward the latter. A mouse, for example, senses only movement, but the feedback given to the user is of a *cursor position*, which is a logical state maintained entirely for the purposes of enabling interactions between the computer and the user. In point-based interactions, the alternative (echoing back movement without showing a cursor) makes little sense. Richer input streams, meanwhile, might tempt designers to skew their feedback more toward unprocessed data, since it may represent a richer visualization (Buxton 1990; Freeman et al. 2009; Wilson and Benko 2010). Although richer, such a representation may not help the user understand cause and effect. Designs can also include both representations, as illustrated by the LucidTouch system (Wigdor, Forlines et al. 2007). LucidTouch shows raw camera data that makes a device appear transparent, but it also includes "touch cursors" to represent the system state. The echo feedback of raw hand images establishes the conceptual model of a device by leading the user to associate each touch cursor to the finger that controls it.

6.9.5 Impoverished Physicality

Modern input devices such as touch screens and in-air gesture systems lack some of the tactile and kinesthetic feedback inherently present in traditional input devices. Such feedback

TABLE 6.3
Potential Causes of Unexpected Behavior and the Source of Feedback That Users Receive to Refute the Causes in Representative Mouse versus Touch Input Systems

Cause of Unexpected Behavior	Feedback Refuting Cause	
	Mouse	Touch
System is nonresponsive	OS: pointer movement	Application
Hardware failed to detect input	HW: activation of button	Application
Input delivered to wrong location	OS: visible pointer	Application
Input does not map to expected function	Application	Application
System is in a particular mode	OS + application: pointer icon	Application
Maximum size reached	OS: pointer moves past edge	Application
Accidental input (arm brushing)	N/A	Application
Overconstrained (too many contacts)	N/A	Application
Stolen capture (second user captures control)	N/A	Application

Note: OS = provided by operating system, HW = provided by hardware.

is an essential element of user experience, especially when users are attempting to understand why a system's response is not as expected. Many commercial mobile devices use audio data to compensate for this problem, with phones that beep or click when a virtual key is pressed.

To understand the role played by feedback, consider Table 6.3, which describes various states of a system and the feedbacks that are provided by either the cursor or the hardware. As is immediately evident, most touch-based platforms shift a great deal of the feedback burden onto the application developer.

However, this difficulty can be addressed by introducing feedback visualizations that represent these specific states and that replace physical responses with visual ones. Such approaches can help to overcome issues of impoverished physicality and semantic feedback, while making touch devices appear more responsive to users (Wigdor et al. 2009).

6.9.6 Snapping Behaviors, Active Haptic Feedback, and Tactile Feedback

Software constraints, such as snapping (Baudisch et al. 2005), often suffice to support a user's tasks without resorting to exotic haptic or tactile feedback mechanisms. Active force or tactile feedback (Burdea 1996) can provide attractive forces for a target or additional feedback for the boundaries of a target, but when evaluated for indirect pointing devices such feedback typically yields little or no performance advantage even for isolated target selections (Akamatsu and Mackenzie 1996; MacKenzie 1995). Such techniques must evaluate selection among multiple targets, because haptic feedback or snapping behavior for one target may interfere with the selection of others (Grossman and Balakrishnan 2005; Oakley, Brewster, and Gray 2001). Visual dominance refers to the tendency of vision to dominate other modalities (Wickens 1992); haptic feedback typically must closely match visual feedback, which limits its utility as an independent modality (Campbell et al. 1999). One promising use of tactile feedback is to improve state transitions, particularly for direct-input devices (Poupyrev and Maruyama 2003; Snibbe and MacLean 2001; Bau et al. 2010).

6.10 KEYBOARDS AND TEXT ENTRY

Typewriters have been in use for over 140 years; the QWERTY key layout dates back to 1868 (Yamada 1980). Those who continue to predict the demise of keyboards would be wise to consider the resilience of this design in the face of the nearly unimaginable orders of magnitude of change that have vastly transformed nearly all other aspects of computing hardware.

Despite the antiquity of the design, QWERTY keyboards are extremely well suited to human performance and, at least for heavy text entry, mechanical keyboards are unlikely to be supplanted by new key layouts, speech recognition technologies, or other techniques any time soon.

Many factors influence typing performance, including key size, key shape, activation force, key travel distance, and the tactile and auditory feedback provided by striking the keys (Lewis, Potosnak, and Magyar 1997), but these well-established design details are not our focus here. We also discuss touch-screen "soft" keyboards, which have some commonality with mechanical keyboards but raise additional design issues, particularly because they demand so much visual attention.

6.10.1 Procedural Memory and the Power Law of Practice

Procedural memory facilitates human performance of complex sequences of practiced movements, such as riding a bike, seemingly without any cognitive effort (Anderson 1980). Procedural memory (which is sometimes informally referred to as muscle memory) is a distinct type of memory that resides below a person's conscious awareness. Procedural memory enables touch typing on a keyboard with minimal attention (Anderson 1980). As a result, users can focus their attention on mental composition and verification of the text appearing on the screen. Dedicated keys or chorded key presses for frequently used commands (hot keys) likewise allow rapid command invocation (McLoone, Hinckley, and Cutrell 2003). The automation of skills in procedural memory is described by the power law of practice:

$$T = aP^b \tag{6.4}$$

Here, T is the time to perform a task, P is the amount of practice, and multiplier a and exponent b are fit to the observed data (Anderson 1980). For a good example of applying the power law of practice to text entry research, see the study by MacKenzie et al. (2001).

Alternative keyboard layouts such as Dvorak offer perhaps a 5% performance gain (Lewis, Potosnak, and Magyar 1997), but the power law of practice suggests this small gain comes at a substantial cost for retraining time. Split-angle ergonomic QWERTY keyboards are close enough to the standard layout that they preserve much of a user's existing skill for typing. They have also been shown to help maintain neutral posture of the wrist and thus avoid ulnar deviation (Honan et al. 1995; Marklin, Simoneau, and Monroe 1997; Smutz et al. 1994), which has been associated with increased pressure in the carpal tunnel (Putz-Anderson 1988; Rempel et al. 1998).

6.10.2 Two-Thumb Mechanical Keyboards

Many designs for cell phones and other handheld devices, such as the RIM Blackberry, offer two-thumb keyboards with QWERTY key layouts. The principal virtue of QWERTY is that common pairs of letters tend to occur on opposite hands. This alternation is a very efficient movement pattern for both standard and two-thumb keyboards, since one hand completes a key press while the other hand moves to the next key (MacKenzie and Soukoreff 2002). A recent study found that two-thumb keyboards offer text entry rates approaching 60 wpm (wpm stands for words per minute) (Clarkson et al. 2005). This suggests that one-handed text entry rates are fundamentally limited due to the serial nature of character entry, despite novel improvements (MacKenzie et al. 2001; Wigdor and Balakrishnan 2003). Word prediction may help, but it also requires overhead for users to monitor and decide whether to use the predictions. Such techniques also require escape mechanisms to allow entry of out-of-vocabulary words.

6.10.3 Touch-Screen Keyboards

Modern multitouch screens enable text entry that is adequate for mobile interaction, thus bringing touch-screen keyboards to the mainstream. But the user experience with a graphical touch-screen keyboard still faces many challenges and shortcomings that may be addressed by future innovations and optimizations (Gunawardana, Paek, and Meek 2010). There have been many studies of touch-screen key sizes (Sears 1991) and of optimal target sizes in general (Vogel and Baudisch 2007), but in practice the key size is more or less dictated by screen dimensions. Graphical keyboards demand significant visual attention because the user must look at the screen to press the correct key. It therefore splits the user's visual attention between the workspace (where text is being inserted) and the graphical keyboard itself. This is particularly problematic on large form factors, such as slates, because the insertion point may be relatively far from the keyboard; if one brushes the screen by accident while typing, for example, one may not even notice that characters are being mistakenly inserted at a new and unintended location. The quality of tactile feedback is poor when compared with a physical keyboard because the user cannot feel the key boundaries. Many graphical keyboards add audible clicks to provide confirmatory feedback, but it is unclear if this actually benefits performance (Sears 1991; Lewis, Potosnak, and Magyar 1997). A graphical keyboard (as well as the user's hand) occludes a significant portion of a device's screen, resulting in less space for the document itself. Furthermore, because the user typically cannot rest his or her fingers in contact with the display (as one can with mechanical keys) and also because the user must carefully keep other fingers pulled back so as to not accidentally touch keys other than the intended ones, extended use of touch-screen keyboards can be fatiguing.

Innovative design hybrids that blend touch or stylus-based typing with stroke gestures have been shown to produce high rates of text entry once the user masters them (Kristensson and Zhai 2004; Zhai and Kristensson 2003; Zhai et al. 2009). It remains unclear if such approaches will achieve widespread adoption.

6.10.4 Handwriting and Character Recognition

Handwriting (even on paper, with no recognition involved) proceeds at about 15 wpm. Thus, a pen is a poor replacement for a keyboard; but, of course, a keyboard is an equally poor replacement for a pen. The specter of recognition arises as soon as one contemplates marking a virtual sheet of paper, but it is important to keep in mind that ink has significant value as a natural data type without recognition: It offers an expressive mix of writing, sketches, and diagrams, and when used to annotate a document concise pen marks or highlights implicitly emphasize the important points in context. Freeform pen input also may help users to generate a breadth of design concepts (Buxton 2007) as opposed to more rigid input mechanisms such as keyboard-based text entry.

Handwriting recognition technology on tablet personal computers has improved markedly over the past decade. Nonetheless, recognition of natural handwriting remains difficult and prone to error for computers, and it demands error correction input from the user. Handwriting recognition works well for short phrases such as search terms

(Hinckley et al. 2007) or for background tasks such as indexing handwritten documents for search, but converting lengthy handwritten passages to error-free text remains a tedious process. Hence, although handwriting recognition is an important enabling technology, in our view pen-operated devices can best avoid the graveyard by emphasizing those user experiences that make minimal demands for recognition and instead emphasize the virtues of ink as a uniquely expressive data type.

In order to make performance more predictable for users, some devices rely on character recognition, which is often implemented as single-stroke (unistroke) gestures (Goldberg and Richardson 1993). Unistroke alphabets attempt to strike a design balance such that each letter is easy for a computer to distinguish yet also straightforward for users to learn (MacKenzie and Zhang 1997). With the widespread adoption of touch-screen keyboards, coupled with the great strides made in handwriting recognition, such approaches have fallen out of favor in most contexts.

6.11 MODALITIES OF INTERACTION

In the search for designs that enhance interfaces and enable new usage scenarios, researchers have explored many input modalities and interaction strategies that transcend any specific type of device.

6.11.1 Bimanual Input

People use both hands to accomplish most real-world tasks (Guiard 1987), but computer interfaces make little use of the nonpreferred hand for tasks other than typing. Bimanual input enables compound input tasks such as navigation/selection tasks, where the user can scroll with the nonpreferred hand while handling the mouse using the preferred hand (Buxton and Myers 1986). This assignment of roles to the hands corresponds with Guiard's kinematic chain theory (Guiard 1987): The nonpreferred hand sets a frame of reference (scrolling to a location in the document) for the action of the preferred hand (selecting an item within the page using the mouse).

Interaction with handheld devices often requires two hands for some tasks. For example, users often hold a device with the nonpreferred hand while the preferred hand performs pinch-to-zoom gestures, taps on small targets, or punches out messages on a soft keyboard. Researchers have explored use of the nonpreferred hand for spatial manipulation by moving or tilting the device (Fitzmaurice and Buxton 1994; Hinckley and Song 2010). This approach leaves the preferred hand free to point at or sketch the content thus revealed.

Some example applications for bimanual input include command selection (Bier et al. 1993; Kabbash, Buxton, and Sellen 1994), drawing tools (Kurtenbach et al. 1997), and virtual camera control and manipulation (Balakrishnan and Kurtenbach 1999; Hinckley et al. 1998). Integrating additional buttons and controls with keyboards to encourage bimanual interaction can also improve the efficiency of some common tasks (MacKenzie and Guiard 2001; McLoone, Hinckley, and Cutrell 2003). Mode switching initiated by the nonpreferred hand, such as by holding down a mode button, has also been demonstrated to be a particularly effective means of changing mode in pen-based interfaces (Li et al. 2005).

6.11.2 Gesture Recognition versus Physics-Based Manipulation

As previously discussed in Section 6.2.1, researchers have found that there does not exist a universal set of natural gestures that users will perform without appropriate affordances (Wobbrock, Morris, and Wilson 2009). The sole exception is physical manipulation: moving an object from one place to another or otherwise changing its position or orientation. Physics-based systems extend this manipulation by mapping inputs to a virtual world governed by Newtonian physics, leading to user experiences described as natural (Agarawala and Balakrishnan 2006; Wilson et al. 2008; Wilson 2009). This approach has been characterized as "reality-based interaction," which advocates the use of naive physics combined with awareness of the body, surrounding environment, and social interaction as the base for successful user interfaces (Jacob et al. 2008). This is in contrast to an approach where gestures are specifically recognized to differentiate system responses, such as assigning functions to particular hand postures (Baudel and Beaudouin-Lafon 1993). Physics-based systems have the advantage that they "feel" like the real world, but they have not yet been demonstrated to scale to enable the range of functions expected of an interactive system. A hybrid model has also been demonstrated, in which shape-based information is used to manipulate a traditional WIMP GUI (Moscovich 2009). Taxonomies of gestures of touch-based computing are provided by Wobbrock, Morris, and Wilson (2009); and Freeman et al. (2009).

6.11.3 Pen and Pen-Based Gesture Input

Pens lend themselves to command gestures analogous to proofreader's marks, such as crossing out a word to delete it. Note that in this example, the gesture integrates the selection of a delete command with the selection of the word to be deleted. Another example is moving a paragraph by circling it and drawing a line to its new location. This integrates the verb, object, and indirect object by specifying the command, extent of text to be moved, and new location for the text (Hinckley, Baudisch et al. 2005; Kurtenbach and Buxton 1991). Marking menus use straight-line gestures along the primary compass directions for rapid command selection (Kurtenbach, Sellen, and Buxton 1993; Zhao and Balakrishnan 2004).

Pen interfaces must decide whether to treat pen strokes as ink content or as gesture commands. Some applications avoid this recognition problem by treating all strokes as commands (Kurtenbach and Buxton 1991), but for a free-form drawing or note-taking application, users need to interleave ink content and command input. The status quo solution presents commands in a toolbar or menu at the edge of the screen; however, this necessitates round-trips between the work area and the command area (Fitzmaurice, Khan, Piek et al. 2003), which become inconvenient in direct proportion to the display size. Pressing a button with the nonpreferred hand is a fast and robust means to switch between ink and gesture modes (Li et al. 2005).

Techniques that automatically distinguish ink and gestures have been proposed, although only for highly restricted gesture sets (Saund and Lank 2003). Punctuation (tapping) has also been explored as a way to both identify and delimit command phrases (LaViola and Zeleznik 2004). A fundamental problem with both these approaches is that the system cannot classify a set of strokes as a gesture or as ink until the user has finished drawing the entire command phrase. This makes it difficult to provide interactive feedback or to prompt the user with available commands before he or she commits to an operation.

Although moving the pen to toolbars at the edge of the screen seems slow on a tablet computer, in practice this round-trip strategy (Fitzmaurice, Khan, Piek et al. 2003) is difficult to improve upon. On a tablet the size of a standard 8.5 × 11–inch sheet of paper, a round-trip requires approximately 1.5 seconds; however, the user can mentally prepare for the next step of the interaction while moving the pen. A locally drawn gesture (such as a straight-line marking menu command) may take less time to articulate, but thinking about what command to select requires additional time unless the task is a routine one. Pressing a button for gesture mode also requires some overhead, as does lifting the pen at the end of the gesture. Also note that performing a sequence of gestures (e.g., tagging words in a document as key words by circling them) requires time to travel between screen locations. The round-trip strategy absorbs this travel time into the round-trip itself, but with gestures this is an extra cost that reduces the benefit of keeping the interaction localized.

Thus, on a tablet-sized device it is difficult to realize substantial time saving just by reducing round-trips. For our hypothetical task of tagging key words in a document, Figure 6.8 illustrates this predicament for average task times drawn from recent studies (Hinckley et al. 2006; Li et al. 2005). The chart shows two successive command selections and assumes that some mental preparation is required before issuing each command. Thus, the potential benefit of pen gestures depends on the sequence of operations as well as the elimination of multiple round-trips, which may be possible with techniques that integrate selection of verb, object, and indirect object (Hinckley, Baudisch et al. 2005; Kurtenbach and Buxton 1991). Localized interaction may also offer indirect benefits by reducing physical effort and by keeping users' visual attention focused on their work (Grossman et al. 2006; Kabbash, Buxton, and Sellen 1994).

6.11.4 Speech, Multimodal Input, and Situated Interaction

Speech has substantial value without recognition. Computers can augment human–human communication across both time and space by allowing users to record, edit, replay, or transmit digitized speech and sounds (Arons 1993; Stifelman 1996; Buxton 1995). Simultaneously recording speech and handwritten annotations is also a compelling combination for human–human communication and collaboration (Levine and Ehrlich 1991). Systems have used microphone input to detect ambient speech and employ this as a cue to optimize interaction with devices (Horvitz, Jacobs, and Hovel 1999; Sawhney

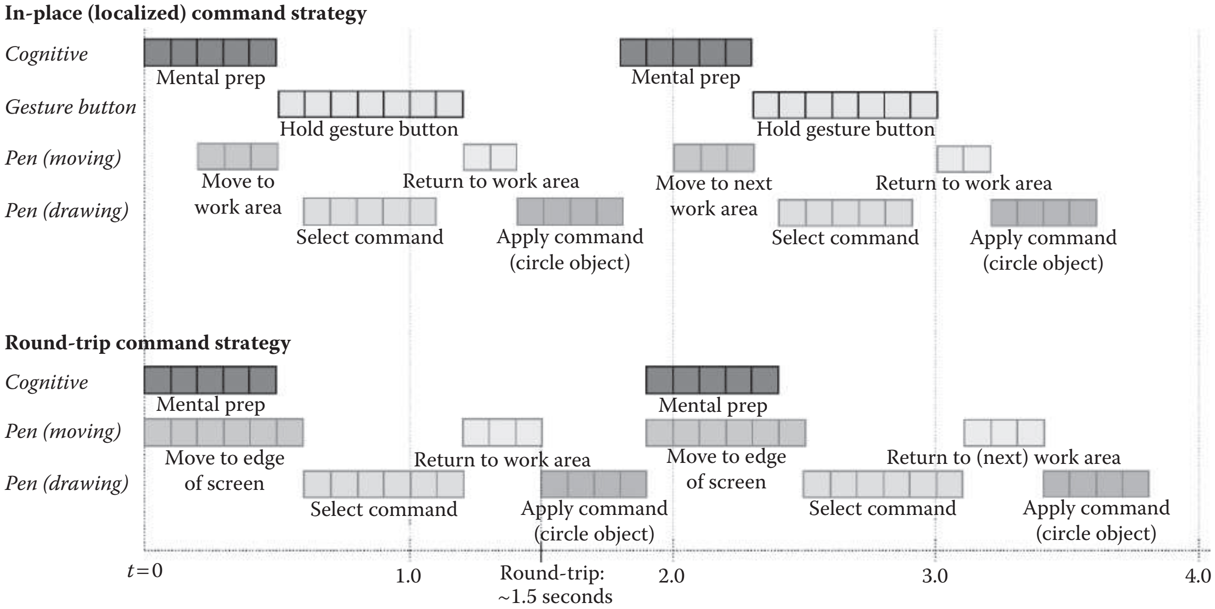


FIGURE 6.8 Time series contrasting the elements of command selection for in-place command selection versus a round-trip to the edge of the screen. These are approximate times for a tablet-sized device (of about 30-cm screen diagonal). (From Hinckley et al. 2006. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montréal, QC: ACM. With permission.)

and Schmandt 2000; Schmandt et al. 2000) or to provide dual-purpose speech, which serves a communicative purpose while also cueing specific actions for a computer (Lyons et al. 2004).

Speech recognition can succeed for a limited vocabulary, such as speaking the name of a person from one's contact list to place a cell phone call; however, error rates increase as vocabulary grows and the complexity of grammar increases, if the quality of the audio signal from the microphone is not good enough, or if users employ out-of-vocabulary words. It is difficult to use speech to refer to spatial locations, so it cannot eliminate the need for pointing (Cohen et al. 1989; Oviatt, DeAngeli, and Kuhn 1997). Keyboard–mouse text entry for the English language is about twice as fast as automatic speech recognition (Karat et al. 1999); furthermore, speaking can sometimes interfere with one's ability to compose text and remember words (Karl, Pettey, and Shneiderman 1993). Finally, speech is inherently nonprivate in public situations. Thus, speech has an important role to play, but claims that speech will soon supplant manual input devices should be considered with skepticism.

Computer understanding of human speech does not enable users to talk to a computer as one would to another person. Doing spoken dialogue well entails *situated interaction*, which goes far beyond speech recognition itself. For computers to embed themselves naturally within the flow of human activities, they must be able to sense and reason about people and their intentions: In any given dialogue, multiple people may come and go; they may interact with the system or with each other, and they may interleave their interactions with other activities such that the computational system is not always in the foreground (Bohus and Horvitz 2010). Dealing with such challenges pushes existing technologies to their limits and, unfortunately, well beyond them at times as well.

6.11.5 Free-Space Gestures and Whole-Body Input

Humans naturally gesture and point using their hands during verbal communication, which has motivated research into freehand gestures, often in combination with speech recognition (Bolt 1980; Hauptmann 1989; Wilson and Oliver 2003; Tse et al. 2008). Cadoz (1994) categorizes hand gestures as semiotic, ergotic, or epistemic. Semiotic gestures, such as thumbs-up, communicate information (Cadoz 1994; Rime and Schiaratura 1991). Ergotic gestures manipulate physical objects. Epistemic gestures are exploratory motions that gather information (Kirsh 1995; Kirsh and Maglio 1994). The interaction literature often focuses on empty-handed semiotic gestures (Freeman and Weissman 1995; Jojic et al. 2000; Maes et al. 1997). A major challenge is to correctly identify when a gesture, as opposed to an incidental hand movement, starts and stops (Baudel and Beaudouin-Lafon 1993; Wilson and Oliver 2003). The lack of deterministic state transitions (Buxton 1990; Vogel and Balakrishnan 2005) can lead to errors of user intent or errors of computer interpretation (Bellotti et al. 2002). Other problems include fatigue from extending one's arms for long periods and the imprecision of pointing at a distance.

By contrast, tangible interaction techniques (Ishii and Ullmer 1997) and augmented devices (Harrison et al. 1998) sense ergotic gestures via a physical intermediary (Hinckley et al. 1998; Zhai, Milgram, and Buxton 1996). The emergence of cameras, cell phones, and tablets augmented with accelerometers, gyroscopes, and other sensors suggest this to be a promising design space.

Whole-body input is also possible, typically utilizing computer vision techniques (Krueger, Gionfriddo, and Hinrichsen 1985). Producing whole-body input by processing the imagery captured by a single camera significantly limits the vocabulary of input to those things that are clearly observable by that single camera, typically to 2D manipulations within the viewing plane (Krueger, Gionfriddo, and Hinrichsen 1985). More recent technologies have augmented this sensor stream with the distance of objects from the camera, enabling more subtle, three-dimensional (3D) interactions. This has been demonstrated by processing two camera images simultaneously and observing binocular disparity (Ko and Yang 1997; Matsushita and Rekimoto 1997). It has also been demonstrated with techniques such as time-of-flight cameras (Wilson 2007), as well as structured light techniques. Commercial products such as Microsoft Kinect provide in-depth information for each pixel, which has the potential to enable richer interaction, such as correcting a projected image to allow for consistently sized projection of objects onto moving surfaces (Wilson 2007).

6.11.6 Background Sensing Techniques

Sensors can enable a mobile device to sense when a user picks up, puts down, looks at, holds, or walks around with the device. These actions give the device information about the context of its use and represent a hidden vocabulary of naturally occurring gestures that people spontaneously exhibit in day-to-day activity. For example, many current smartphones and slate devices employ a tilt sensor to interactively switch the display between portrait and landscape formats, as well as to automatically save photographs in the correct orientation (Hinckley, Pierce et al. 2005; Hinckley et al. 2000). Here, the sensor allows the device to adapt its behavior to a user's needs rather than requiring the user to take extra steps to control the display format and photograph orientation (Buxton 1995).

Sensors can also be embedded in the environment. When one walks into a modern department store, no explicit command is required to open the doors: The doors sense motion and automatically open. Researchers are investigating new ways to leverage such contextual sensing to enrich and simplify human interaction with devices and digital environments (Abowd and Mynatt 2000; Schilit, Adams, and Want 1994).

6.11.7 Projector-Based Augmented Reality

Direct interaction can be extended beyond screens by projecting imagery onto real-world physical objects. Such a projection is normally coupled with a sensor that enables sensing of the movement of the object on which the imagery is projected, of users' hands within the projection area, or both. Such sensing has been enabled through projection of imagery onto input devices, thereby transforming indirect multitouch input devices into direct ones (Dietz and Leigh 2001). More typical is the use of a camera with a viewing area that overlaps the projection area (Ullmer and Ishii 1997; Matsushita and Rekimoto 1997; Klemmer et al. 2001). This has been envisioned as a lightbulb capable of high-resolution information display and of sensing user interaction with that display (Underkoffler and Ishii 1998). The technique has been used to augment physical objects with digital information, such as overlaying instructions for mechanical operations and augmenting data ports with network traffic information (Raskar et al. 2005). The technique has also been augmented by sensing manipulations of the projector itself, such as rolling the projector to zoom the user interface (UI) (Forlines et al. 2005; Cao and Balakrishnan 2006), and by allowing multiple projectors to overlap one another envisioning multiuser interaction scenarios such as sharing calendars (Cao, Forlines, and Balakrishnan 2007). Small projectors have also been mounted on other input devices, such as a stylus or a mouse, to augment those interactions (Song et al. 2009; Song et al. 2010).

6.11.8 Direct Muscle-Based Input and Brain–Computer Interfaces

Traditional input devices can be thought of as secondary sensors, in that they sense a physical action that is the consequence of cognition and muscle movements. An alternative approach is to attempt primary sensing by detecting brain activity and muscle movements directly. Muscle sensing is accomplished through electromyography, a technique previously employed for measuring muscular activity or controlling prosthetics. Saponas et al. (2009) demonstrated its use to enable sensing of muscle activation as fine-grained as detecting and identifying individual fingers, and used it in combination with touch-screen input to provide a richer data stream (Benko et al. 2009). Brain–computer interfaces (BCIs) typically employ electroencephalography (EEG) or functional near-infrared spectroscopy to detect input. Projects have used the technique to detect workload and user engagement (Hirshfield, Chauncey et al. 2009) in order to conduct usability studies, as well as to explore the possible dynamic adaptation of user interfaces based on such metrics (Hirshfield, Solovey et al. 2009). Such work remains in its infancy, but it appears to hold great promise (particularly as assistive technologies for users suffering from devastating injuries or significant physical limitations) with the improvement of sensing and signal-processing techniques.

6.12 FORM FACTORS FOR DIRECT (AND INDIRECT) INTERACTION

The form factor of a device is a function of both technology and the envisioned context of use of that device. This context extends, among other things, to the tasks a user is likely to want to perform, the physical environment in which the device is likely to be found, and the other devices that populate the surrounding ecosystem. Mobile phones and desktop computers enable some overlapping tasks and may even be put to complementary uses by the same user at the same time for some tasks, but for each device the focus of its design is on different usage contexts.

Here, we provide some example devices to sketch the terrain and show examples of how hardware form factor, input capabilities, and display size combine to create unique user experiences and how each of these classes of devices in turn offers unique interaction properties and hence new user experiences. In practice, these user experiences may be realized only if designers focus on the depth of an experience by designing for the context of use of a device, which perhaps necessitates foregoing breadth of solution.

6.12.1 Handheld Devices and Smartphones

Handheld devices promise quick access to information in many contexts, since they are easily carried in a pocket or purse. Early on, these were dubbed "personal digital assistants" (PDAs), whose purpose was to track contacts, make notes, and keep a calendar. Modern versions of PDAs are highly valued not only for the information they contain but also because they are persistently online, promising instant access to real-time data with e-mail, web browsing, and of course telephony as core functions. Their convenience of use often overcomes the limitations of these devices such as their tiny screens and inefficient text entry. A symptom of the success as well as the inadequacies of such devices is the caveat *sent from my mobile device* that pervades automatic e-mail signatures on smartphones, as if the messages were crafted by a carpenter who blames his tools for the shoddy construction.

One of the first real successes for handheld computing was the PalmPilot. The PalmPilot was designed to fit into a human being's shirt pocket. The screen was a resistive touch digitizer, enabling interaction using either a finger or a plastic stylus. Although one-handed interaction was possible in some situations, the form factor was primarily intended to be held in the nonpreferred hand while the preferred hand interacted with the screen via a finger or stylus. The PalmPilot's designers recognized that they could enable faster text entry and more precise pointing by using the stylus, which was sheathed in the bezel of the device.

Although the PalmPilot's form factor was ostensibly similar to that of the Apple iPhone, the iPhone replaced the PalmPilot's resistive screen with a highly sensitive capacitive touch screen enabling multitouch interactions such as the now ubiquitous pinch gesture (Krueger, Gionfriddo, and Hinrichsen 1985). Perhaps even more important than the multitouch capability was the soft-touch contact sensing afforded by the capacitive touch screen. Soft-touch sensing enabled more precise pointing as well as more pervasive use of dragging for interactive manipulations such as panning, zooming, and scrolling with inertia in a way that was difficult to support well on the prior generation of resistive touch screens. Of course, use of modern processors, graphics hardware, and well-designed animated feedback also made the interactions clear and rewarding for users to experience.

Many handheld devices are designed for use predominantly with a single hand, with the device secured by the palm while the thumb interacts with the screen. Because thumb input is significantly less precise than stylus input, a specialized user interface for the thumb should be considered (Karlson 2007). For example, zooming user interfaces enable interaction with large amounts of data on a small screen while using only the thumb (Karlson 2005). Ironically, however, the iPhone's iconic pinch-to-zoom gesture requires two hands (one hand to hold the device while the other hand articulates the pinch gesture itself). Touching the screen while tilting a handheld device offers an alternative for one-handed continuous zooming (Hinckley and Song 2010).

Both the PalmPilot and the iPhone do away with a physical keyboard in favor of a larger screen. As always, this is a trade-off supporting one use case (viewing a variety of data sources) over another (entering text for documents, e-mails, etc.). A variety of devices make the other choice, opting for a physical keyboard. A now classic form factor is the "clamshell"; it provides a physical keyboard attached to the bottom of a display, which affords use either by typing with two thumbs or by placing the device on a table. A noteworthy early device in this space is the Psion Series 5 device (Figure 6.9). The Psion included a physical keyboard, along with a resistive touch digitizer. Its hinging mechanism was a clever touch of its industrial design: The display slides forward as the device opens, such that the base of the device cantilevers the display. This enables the user to interact with the angled touch screen without it flopping around or toppling the device.

As digital devices become physically smaller, the displays and input mechanisms they offer shrink in size. Considerable effort was once devoted to supporting web browsing in limited screen space (e.g., Buyukkokten, Garcia-Molina, and Paepcke 2001; Trevor et al. 2001), but much of this effort now seems to have been obviated by pinch-to-zoom on touch-sensitive displays plus the emergence of mobile themes for many websites. Techniques to make small displays virtually larger include peephole displays (Fitzmaurice 1993; Yee 2003), transparent overlays (Harrison et al. 1995; Kamba et al. 1996), and use of on-screen visuals to suggest the location of off-screen objects (Baudisch and Rosenholtz 2003). The latter, for example, is one of the principal techniques embraced by the Windows Phone 7 Metro UI design, by having a typography and UI layout that suggests the presence of additional options beyond the periphery of the current view.

Physical manipulations such as tilting that use the device itself as an interface seem particularly well suited to small devices (Harrison et al. 1998; Hinckley et al. 2000; Rekimoto 1996). Tiny, bright, and inexpensive laser or light-emitting diode (LED) projectors lie just around the corner; progress on computer vision techniques suggests that interactive projection may allow small devices to project large displays and sensing surfaces (Raskar et al. 2004; Wilson 2005), but brightness and power consumption remain significant obstacles to the widespread adoption of projection displays. Ways to interact with even smaller devices, such as pendants, rings, watches, or even pointlike sensors/emitters, have also been considered (Ni and Baudisch 2009).

6.12.2 Slates and Tablets

A number of compelling direct-input devices are starting to populate the long-standing void between handheld devices

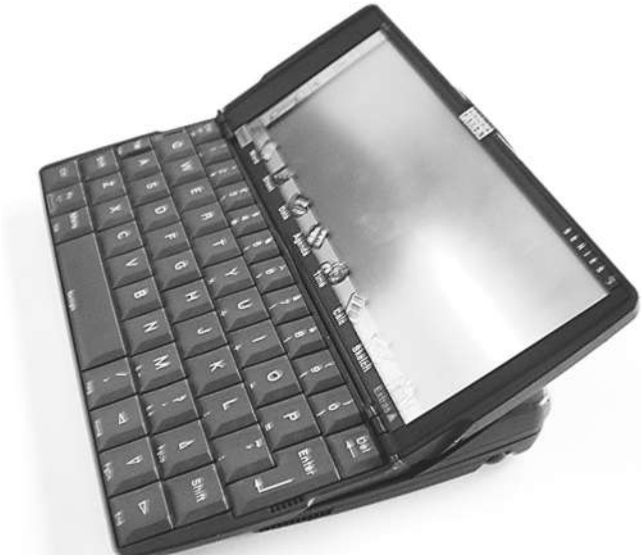


FIGURE 6.9 The Psion Series 5 handheld device, circa 1995. The resistive touch screen supports stylus or touch input. Note the counterweight on the bottom part of the clamshell that prevents the device from tipping over when one interacts with the screen.

and larger devices such as drafting boards, tabletop devices, and wall-mounted displays. Of course, laptops have existed in this category for many years, but these have essentially been caricatures of the traditional desktop interface.

Slates, e-readers, dual-screen booklets, and other emerging form factors such as flexible devices support new and specialized uses that stretch the boundaries of traditional computing experiences (Holman and Vertegaal 2008). The new usage contexts afforded by these devices, many of which emphasize direct interaction by touch or pen, or both, are placing new demands on input and interaction techniques, signaling an opportunity for innovation in both industry and academia. As our colleague Bill Buxton (www.billbuxton.com, accessed on September 1, 2010) writes, "Slate computers and e-readers represent a new class of digital appliance—one targeted for casual use. With the growth of this market will emerge a new and long overdue approach to interaction—one that is in keeping with the casual intent and context of such usage and which will complement, rather than replace, interfaces that support more formal and structured activities."

Even within this emerging category of devices, there remains much room for differentiation and specialization. Devices such as the Amazon Kindle focus on recreational reading. Devices such as the Apple iPad support reading as one of its many dedicated applications. Both have been successful in the marketplace. From the literature we also know that in many contexts, reading tends to occur in conjunction with writing, such as a student studying a textbook or an information worker bringing together documents to research a topic (O'Hara and Sellen 1997; Adler and van Doren 1972). Devices such as the iRex iLiad, which supports pen annotation on top of e-book content, and the Courier project (Microsoft Courier 2010), which explores a dual-screen booklet form factor supporting both pen and touch inputs, hint at further developments to come. Digital annotation (Price, Schilit, and Golovchinsky 1998) and dual-screen form factors (Chen et al. 2008; Hinckley et al. 2009) have also been explored in the research literature.

Such midsize devices place new demands on interaction design, even when they support similar or identical touch-screen interfaces to analogous handheld devices. For example, one reason that the Apple iPad must have such a wide bezel is that it would otherwise be unclear how to grasp the device without accidentally triggering operations on its touch screen. Our personal experience with touch-screen tablets of this sort suggests that one is nonetheless still far more likely to touch things by accident than on a corresponding handheld touch interface. This once again illustrates how even with a subtle shift in the underlying form factor a desirable property of an input modality—in this case, that one can just lightly touch the screen to initiate an action—can become a liability: If you just lightly touch the screen *by accident*, you will initiate some (undesired) action. New but related problems also crop up, such as the need to ameliorate accidental palm contact when writing on such a device with a pen, which do not manifest themselves on smaller handheld form factors.

6.12.3 Desktops

Rumors regarding the death of the desktop computer are greatly exaggerated. After 40 years of iterative design, the personal computer, driven by a mouse and keyboard and controlling a WIMP GUI, continues to evolve. Pointing continues to be the primary unit of interaction, where the mouse's fundamental properties make it difficult to beat (Balakrishnan et al. 1997). Another primitive of interaction, that is, scrolling, typically also has a dedicated input device in the form of the scroll wheel. Augmented desktops with additional peripheral displays for direct pen input on horizontal and vertical surfaces have also been explored (Morris, Brush, and Meyers 2008).

Desktops can be expanded with multiple monitors. Research has shown that users have a strong tendency to partition tasks between discrete physical displays; for example, users often dedicate monitors to particular tasks or types of applications (Grudin 2001). Monitor bezels influence the way in which users arrange application windows (Hutchings and Stasko 2004) and lead to discontinuities in information displays (Tan and Czerwinski 2003). Thus, the presence of a bezel between multiple monitors can be both a benefit and a hindrance depending on a user's task; but clearly, one large display is not necessarily superior to multiple smaller displays separated by bezels. Augmented desktops with additional peripheral displays for direct pen input on horizontal and vertical surfaces have also been explored (Morris, Brush, and Meyers 2008).

The potential negative consequences of having bezels between displays often can be mitigated by careful handling of inputs and appropriate display of graphical feedback. For example, researchers have explored applications that are aware of the seams and can adjust their layout accordingly (Mackinlay and Heer 2004), as well as windowing systems that reposition and duplicate dialog boxes (Hutchings and Stasko 2005). Interaction techniques have also been proposed to account for the physical layout of the monitors. These have included the mouse ether technique in which the physical gaps between monitors are echoed in a virtual motor space that the mouse cursor moves through, leading to better correspondence between input and output in accordance with users' expectations (Baudisch et al. 2004). Researchers have experimented with assigning each monitor its own mouse cursor (Benko and Feiner 2005) as well as the use of head tracking to facilitate jumps of the mouse cursor between distant monitors (Ashdown, Oka, and Sato 2005).

6.12.4 Multitouch Tables and Screens

Screen size is an obvious delimiter between different categories of devices. But so too is screen orientation: Research has demonstrated that large touch screens mounted horizontally afford uses distinct from screens mounted vertically.

Extensive study of tabletop interaction has unearthed many issues and techniques unique to the large horizontal form factor. Most relate to multiuser usage scenarios, which a large horizontal screen naturally affords because multiple users can sit while facing one another (Shen, Everitt, and Ryall 2003). Although conducive to multiuser interaction, such an arrangement creates issues of content orientation, since an on-screen object oriented correctly for one user is necessarily oriented incorrectly for other users (Shen et al. 2004). Systems can mitigate this problem by altering the view seen from each side of the table (Matsushita et al. 2004) by using head-worn displays as the sole display mechanism (Agrawala et al. 1997) or in combination with other information displays, such as projecting directly onto an input device (Benko, Ishak, and Feiner 2004).

Sharing a single tabletop also leads to issues. Users have been observed to implicitly and fluidly delineate a tabletop display into multiple territories, personal, shared, and storage (Scott et al. 2004), varying by the size of the table (Ryall et al. 2004). It is not clear how much of this separation is due simply to issues of comfort of reach, as described by anthropometricists as the *kinetosphere* (Toney and Bruce 2006), and how much is dictated by the mores of social distance as studied in the field of *proxemics* (Ballendat, Marquardt, and Greenberg 2010). Viewing information horizontally and from different sides has been shown to lead to perceptual differences (Wigdor, Shen et al. 2007) and to improve visual search efficiency (Forlines et al. 2006).

In addition to social and psychological factors, multiuser interaction with a single display breaks many existing UI paradigms, such as persistent modes, undo stacks, toolbars, and paint palettes that set a global mode in the interface when a user taps on the control. If your system is one of the many that does not offer user differentiation for its contacts and your interface design includes such elements, then your design is in trouble. Hence, cooperative tools for multiple users must often seek clever design solutions that avoid all global modes in an interface.

This is mitigated somewhat by technologies such as the DiamondTouch, which can identify users and thereby provide a specific mode for each distinct user (Dietz and Leigh 2001). Addressing these and other similar issues, researchers have designed many user interface elements specifically for shared horizontal form factors, such as shared versus individual controls (Morris et al. 2006), methods for supporting multiple off-axis viewpoints (Hancock and Carpendale 2007), multitouch content reorientation methods (Hancock et al. 2006), and text entry (Hinrichs et al. 2007). Researchers have also examined altering the form factor and have demonstrated differences in user behavior based on screen size (Ryall et al. 2004). Hartmann et al. (2009) demonstrated the augmentation of a large (4′ × 6′) tabletop with physical controls, such as multiple keyboards and mice, as well as other physical objects.

6.12.5 Vertical Displays and Very Large Displays

Mounting on a wall enables form factors, which include large screens. Large displays lend themselves to collaboration and the sharing of information with groups (Funkhouser and Li 2000; Swaminathan and Sato 1997), as well as giving a substantial physical presence to virtual activities (Buxton et al. 2000; Trimble, Wales, and Gossweiler 2003).

The digital whiteboard is a classic example of a large vertically oriented display (Elrod et al. 1992; Moran, Chiu, and Melle 1997; Pederson et al. 1993). Many digital whiteboards mix the physical and the digital by tracking the position of real markers as a user strokes on the board. Many boards are augmented with projection on the writing area, allowing users to interact with applications while still making digital or physical marker strokes. Many digital whiteboards' reliance on projectors limits their resolution, enabling a large but coarse information display.

Alternative technologies can enable large, high-resolution displays, although such displays typically require the tiling of either projectors or individual displays. Each of these technologies has inherent limitations. Tiled screens currently require a bezel (nondisplay area), which creates a grid of tiles that interferes with interactions such as direct-touch dragging of items across screens (Ball and North 2005). By contrast, in projected imagery, the lack of physical separation of the display area creates areas of nonuniform brightness and color even after careful calibration (Surati 1999). Despite these issues, large high-resolution displays have been demonstrated to aid spatial tasks (Tan et al. 2003), improve and reduce gender effects in navigation (Czerwinski, Tan, and Robertson 2002), provide utility in sense making (Andrews, Endert, and North 2010), and influence visual search and steering tasks (Bi, Bae, and Balakrishnan 2010).

Direct input on wall-mounted displays is commonplace, but the constant physical movement required can become burdensome. Hybrid approaches that mix absolute and relative direct pointing have been proposed (Forlines, Vogel, and Balakrishnan 2006), as have methods for bringing distant targets closer to the user (Baudisch, Bederson, and Zierlinger 2002; Bezerianos and Balakrishnan 2005; Collomb et al. 2005). Interaction-at-a-distance can be afforded via indirect interaction techniques, such as touch input on a nearby tablet device (Nacenta et al. 2005; Malik, Ranjan, and Balakrishnan 2005), eye tracking (Bolt 1981; Shell et al. 2004), or by pointing at the display using a physical device (Bolt 1980; Wilson and Shafer 2003) or the hand itself (Ko and Yang 1997; Nickel and Stiefelhagen 2003; Vogel and Balakrishnan 2005). Even when a user is close to a large display, interacting with portions of the display that are out of view or beyond arm's length raises challenges (Bezerianos and Balakrishnan 2005; Khan et al. 2005).

6.12.6 Federations of Form Factors: A Society of Devices

As form factors continue to evolve and multiple form factors per user become more affordable, each of us is likely to increasingly interact with our digital information and with one another through an ever-broadening array of devices, each suited to the contexts described till now. One early prediction of such an ecology of devices was Weiser's vision

(Weiser 1991) of ubiquitous computing, comprising tabs, pads, and boards, which foreshadowed the billion-dollar markets for smartphones, e-readers, and slate devices.

Displays of various sizes support different activities and social conventions; one of the principal challenges of ubiquitous computing is finding techniques that make it easy for users to work within a digital ecology that supports a range of tasks spanning multiple computers, displays, and interactive surfaces. Several projects have probed how to use small displays as adjuncts to large ones (Myers, Lie, and Yang 2000; Myers, Stiel, and Gargiulo 1998; Rekimoto 1998; Streitz et al. 1999), allowing simultaneous interaction with private information on a personal device and a shared or public context on a larger display.

What makes such federations of devices fascinating are not so much the individual devices themselves but rather the virtual bridges that span the spaces between them. Wireless networking is the technology that will perhaps most severely disrupt traditional approaches to HCI and computing in general in the coming years, because it breaks down barriers between devices and enables a new society of devices that fill specific roles and can still coordinate their activities (Fitzmaurice, Khan, Buxton et al. 2003; Want and Borriello 2000). Users need techniques that allow them to access and share information across the boundaries of individual devices, as well as to dynamically bind together multiple devices and displays to accomplish their tasks (Hinckley et al. 2004; Rekimoto et al. 2003). Such interaction techniques inherently involve multiple individuals, and thus they must also consider how people use physical proximity and relative body orientation (Deasy and Lasswell 1985; Hinckley et al. 2004; Sommer 1965).

Thus, the interaction designer must consider the full range of scale for display sizes and form factors that may embody an interaction task, as well as the interactions between different types of devices. How can interaction migrate from watches, cell phones, handheld devices, and tablets all the way up to desktop monitors, digital whiteboards, and interactive wall-sized displays? There is, as yet, no simple answer to such questions.

6.13 TRENDS AND OPPORTUNITIES FOR INPUT

The designer of an interactive system should take a broad view of input and consider not only traditional pointing techniques and GUI widgets but also issues such as search strategies to access information in the first place, sensor inputs that enable entirely new data types, and synthesis techniques to extract meaningful structure from data. Good search tools may reduce the many inputs needed to manually search and navigate file systems. Knowledge work requires integration of external information from web pages or databases (Yee et al. 2003) as well as reuse of personal information from documents, e-mails, and other content authored or viewed by a user (Dumais et al. 2003; Lansdale and Edmonds 1992). Unified full-text indexing allows users to quickly query their personal information across multiple information silos and can present information in the context of memory landmarks such as the date on which a message was sent or the application that was used to create a document (Cutrell et al. 2006).

Sensor inputs, such as those from accelerometers, gyroscopes, magnetometers, proximity sensors, and other sensors, are now widespread in mobile devices. Cell phones and low-power radios for wireless networking can sense their location or proximity to other devices via analyses of signal strengths (Bahl and Padmanabhan 2000; Krumm and Hinckley 2004). As another example, attempting to type a secure password on a mobile phone keypad quickly convinces one that biometric sensors or some other convenient means for establishing identity is essential to support secure interactions without completely frustrating users. Such sensors could make services such as personalization of interfaces much simpler. Research has already demonstrated how on-screen fingerprint sensing can also enable high-precision touch-screen interaction (Holz and Baudisch 2010).

The need to extract models and synthesize structure from large quantities of low-level inputs suggests that data mining and machine learning techniques will become important adjuncts to interaction (Fails and Olsen 2003; Fitzmaurice, Balakrisnan, and Kurtenbach 1999; Horvitz et al. 1998). Whenever a system considers automatic actions on behalf of a user, however, an important design principle in the face of uncertainty is to "do less, but do it well" (Horvitz 1999). A key challenge for input will be representation of these structures: As streams of sensors collect information, how to make this information available to applications so that their responses can be designed and customized is as yet unclear, but a probabilistic framework for handling input events may be the right general approach (Schwarz et al. 2010). Applying raw inputs to a physics engine is one extreme in the spectrum of possible approaches (Wilson et al. 2008). But we could argue that it is an equally extreme approach to reduce inputs to a lowest common denominator, as is common practice in modern toolkits. Somewhere in between, perhaps, lies a more nuanced solution to the breadth–depth dichotomy that carries a meaningful representation and framework that leverages the specific and deep aspects of individual platforms while also enabling broad solutions of very high quality that embrace the wide array of form factors comprising the modern society of devices.

We must make substantial progress in all these areas to advance human interaction with technology. The forms and capabilities of these and other technologies will continue to advance, but human senses and cognitive skills will not. We will continue to interact with computers using our hands and physical intermediaries, not necessarily because our technology requires us to do so but because touching, holding, and moving physical objects is the foundation of the long evolution of tool use in the human species (Wilson 1998).

Moore's law is often cited as an economic rule of technology, which states that the amount of processing power available at a given price will double every 18 months. A complement to and an expansion of this law is that all elements of device construction, design, electronics, and manufacture have been becoming and will continue to become less expensive over time. A natural consequence of this trend is the society of devices we have described, as more form factors become realizable at lower costs. Many predicted the end of the digital camera market as mobile phone cameras improved. Instead, the digital single-lens reflex camera market has grown significantly alongside the market for ever-improving camera-equipped phones. Likewise, the e-reader market, including slates such as the Amazon Kindle and the Apple iPad, has seen significant expansion despite the availability of Kindle and iBook software free of charge for smartphones and slates. As (bureaucrats? shapers? socialites? framers? founding fathers?) of this society of devices, designers will be well served to keep a broad perspective, taking into consideration many of the issues that we have addressed in this chapter, and thereby avoid the pitfalls of meaningless consistency, hopefully in favor of good design.

REFERENCES

Abowd, G., and E. Mynatt. 2000. Charting past, present, and future research in ubiquitous computing. *ACM Trans Comput Hum Interact* 7(1):29–58.

Accot, J., and S. Zhai. 1997. Beyond Fitts' law: Models for trajectory-based HCI tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Accot, J., and S. Zhai. 2001. Scale effects in steering law tasks. In *Proceedings of CHI '01 ACM Conference on Human Factors in Computing Systems*, 1–8. New York, NY, USA: ACM.

Adler, M. J., and C. van Doren. 1972. *How to Read a Book*. New York: Simon and Schuster.

Agarawala, A., and R. Balakrishnan. 2006. Keepin' it real: Pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montréal, QC: ACM.

Agrawala, M., A. C. Beers, I. McDowall, B. Frohlich, M. Bolas, and P. Hanrahan. 1997. The two-user Responsive Workbench: Support for collaboration through individual views of a shared space. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*. New York: ACM Press/Addison-Wesley Publishing Co.

Akamatsu, M., and I. S. Mackenzie. 1996. Movement characteristics using a mouse with tactile and force feedback. *Int J Hum Comput Stud* 45:483–93.

Anderson, J. R. 1980. Cognitive skills. In *Cognitive Psychology and Its Implications*. San Francisco: W. H. Freeman.

Andrews, C., A. Endert, and C. North. 2010. Space to think: Large high-resolution displays for sensemaking. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Apitz, G., and F. Guimbretière. 2004. CrossY: A crossing-based drawing application. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. Santa Fe, NM: ACM.

Appert, C., M. Beaudouin-Lafon, and W. Mackay. 2004. Context matters: Evaluating interaction techniques with the CIS model. Paper read at Proceedings of HCI 2004, Leeds, UK.

Arons, B. 1993. SpeechSkimmer: Interactively skimming recorded speech. In *Proceedings of the 6th Annual ACM Symposium on User Interface Software and Technology*. Atlanta, GA: ACM.

Ashdown, M., K. Oka, and Y. Sato. 2005. Combining head tracking and mouse input for a GUI on multiple monitors. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. Portland, OR: ACM.

Bahl, P., and V. Padmanabhan. 2000. RADAR: An In-Building RF-based user location and tracking system. Paper read at IEEE 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000).

Balakrishnan, R., T. Baudel, G. Kurtenbach, and G. Fitzmaurice. 1997. The Rockin'Mouse: Integral 3D manipulation on a plane. Paper read at ACM CHI 1997 Conference on Human Factors in Computing Systems, New York.

Balakrishnan, R., and K. Hinckley. 1999. The role of kinesthetic reference frames in two-handed input performance. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*. Asheville, NC: ACM.

Balakrishnan, R., and G. Kurtenbach. 1999. Exploring bimanual camera control and object manipulation in 3D graphics interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*. Pittsburgh, PA: ACM.

Ball, R., and C. North. 2005. Effects of tiled high-resolution display on basic visualization and navigation tasks. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. Portland, OR: ACM.

Ballendat, T., N. Marquardt, and S. Greenberg. 2010. *Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment*. Calgary, AB: University of Calgary.

Bau, O., I. Poupyrev, A. Israr, and C. Harrison. 2010. TeslaTouch: Electrovibration for touch surfaces. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. New York: ACM.

Baudel, T., and M. Beaudouin-Lafon. 1993. Charade: Remote control of objects using hand gestures. *Commun ACM* 36(7):28–35.

Baudisch, P., E. Cutrell, D. Robbins, M. Czerwinski, P. Tandler, B. Bederson, and A. Zierlinger. 2003. Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-operated Systems. In *Proceedings of Interact 2003*, 57–64. Switzerland: Zurich.

Baudisch, P., E. Cutrell, K. Hinckley, and A. Eversole. 2005. Snap-and-go: Helping users align objects without the modality of traditional snapping. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Baudisch, P., E. Cutrell, K. Hinckley, and R. Gruen. 2004. Mouse ether: Accelerating the acquisition of targets across multi-monitor displays. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*. Vienna, Austria: ACM.

Baudisch, P., and R. Rosenholtz. 2003. Halo: A technique for visualizing off-screen objects. Paper read at CHI '03.

Bellotti, V., M. Back, W. Keith Edwards, R. E. Grinter, A. Henderson, and C. Lopes. 2002. Making sense of sensing systems: Five questions for designers and researchers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*. Minneapolis, MN: ACM.

Benko, H., and S. Feiner. 2005. Multi-monitor mouse. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. Portland, OR: ACM.

Benko, H., E. W. Ishak, and S. Feiner. 2004. Collaborative mixed reality visualization of an archaeological excavation. Paper read at Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on, 2–5 Nov. 2004.

Benko, H., S. Izadi, A. D. Wilson, X. Cao, D. Rosenfeld, and K. Hinckley. 2010. Design and evaluation of interaction models for multi-touch mice. In *Proceedings of Graphics Interface 2010*. Ottawa, ON: Canadian Information Processing Society.

Benko, H., T. Scott Saponas, D. Morris, and D. Tan. 2009. Enhancing input on and above the interactive surface with muscle sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. Banff, AB: ACM.

Benko, H., A. D. Wilson, and P. Baudisch. 2006. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montréal, QC: ACM.

Bezerianos, A., and R. Balakrishnan. 2005. The vacuum: Facilitating the manipulation of distant objects. Paper read at CHI '05.

Bi, X., S.-H. Bae, and R. Balakrishnan. 2010. Effects of interior bezels of tiled-monitor large displays on visual search, tunnel steering, and target selection. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Bier, E. A., M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. 1993. Toolglass and magic lenses: The see-through interface. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*. Anaheim, CA: ACM.

Bohus, D., and E. Horvitz. 2010. On the challenges and opportunities of physically situated dialog. Paper read at AAAI Fall Symposium on Dialog with Robots, Arlington, VA.

Bolt, R. 1980. Put-that-there: Voice and gesture at the graphics interface. *ACM SIGGRAPH Comput Graph* 14(3):262–70.

Bolt, R. A. 1981. Gaze-orchestrated dynamic windows. In *Proceedings of the 8th Annual Conference on Computer Graphics and Interactive Techniques*. Dallas, TX: ACM.

Bowman, D., E. Kruijff, J. LaViola, and I. Poupyrev. 2004. *3D User Interfaces: Theory and Practice*. Boston, MA: Addison-Wesley.

Brandl, P., C. Forlines, D. Wigdor, M. Haller, and C. Shen. 2008. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. Paper read at Proceedings of AVI '08 Conference on Advanced Visual interfaces.

Brandl, P., J. Leitner, T. Seifried, M. Haller, B. Doray, and P. To. 2009. Occlusion-aware menu design for digital tabletops. Paper read at CHI 2009 Extended Abstracts.

Britton, E., J. Lipscomb, and M. Pique. 1978. Making Nested Rotations Convenient for the User. *Comput Graph* 12(3):222–7.

Brown, E., W. Buxton, and K. Murtagh. 1990. Windows on tablets as a means of achieving virtual input devices. Paper read at Interact '90, Amsterdam.

Burdea, G. 1996. *Force and Touch Feedback for Virtual Reality*. New York: John Wiley & Sons.

Buxton, W. 1986a. There's more to interaction than meets the eye. In *User Centered System Design: New Perspectives on Human–Computer Interaction*, ed. D. Norman and S. Draper. Hillsdale, NJ: Lawrence Erlbaum Associates.

Buxton, W. 1986. Chuncking and phrasing and the design of human-computer dialogues. In *Readings in Human–Computer Interaction: Towards the Year 2000*, ed. R. Baecker, J. Grudin, W. Buxton, and S. Greenberg, 475–80. San Fransisco, CA: Morgan Kaufmann.

Buxton, W. 1990a. A three-state model of graphical input. In *Proceedings of the IFIP TC13 Third Interational Conference on Human–Computer Interaction*. North-Holland Publishing Co.

Buxton, W. 1990b. Three-state model of graphical input. In *Human–Computer Interaction—INTERACT '90*, ed. D. Diaper, 449–56. Amsterdam: Elsevier Science Publishers B. V. (North-Holland).

Buxton, W. 1995a. Integrating the Periphery and Context: A New Taxonomy of Telematics. Paper read at Proceedings of Graphics Interface '95, Quebec City, QC.

Buxton, W. 1995b. Speech, language and audition. In *Readings in Human–Computer Interaction: Toward the Year 2000*, ed. R. Baecker, J. Grudin, W. Buxton, and S. Greenberg. Somerville, MA: Morgan Kaufmann Publishers.

Buxton, W. 1995c. Touch, gesture, and marking. In *Readings in Human–Computer Interaction: Toward the Year 2000*, ed. R. Baecker, J. Grudin, W. Buxton, and S. Greenberg. Somerville, MA: Morgan Kaufmann Publishers.

Buxton, W. 1995. Chunking and phrasing and the design of human-computer dialogues. In *Human-computer interaction*, ed. R. M. Baecker, J. Grudin, W. Buxton, and S. Greenberg, 494–9. San Francisco, CA: Morgan Kaufmann Publishers Inc.

Buxton, B. 2007. *Sketching User Experiences: Getting the Design Right and the Right Design*. San Francisco, CA: Morgan Kaufman.

Buxton, W., G. Fitzmaurice, R. Balakrishnan, and G. Kurtenbach. 2000. Large displays in automotive design. *IEEE Comput Graph Appl* 20(4):68–75.

Buxton, W., R. Hill, and P. Rowley. 1985. Issues and techniques in touch-sensitive tablet input. *Comput Graph* 19(3):215–24.

Buxton, W., and B. Myers. 1986a. A study in two-handed input. *SIGCHI Bull* 17(4):321–6.

Buxton, W., and B. Myers. 1986b. A study in two-handed input. In *ACM CHI 1986 Conference on Human Factors in Computing Systems*. New York: ACM.

Buyukkokten, O., H. Garcia-Molina, A. Paepcke, and T. Winograd. 2000. Power browser: efficient Web browsing for PDAs. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (CHI '00), 430–437. New York, NY, USA: ACM.

Cadoz, C. 1994. *Les Realites Virtuelles*. Dominos, Flammarion.

Campbell, C., S. Zhai, K. May, and P. Maglio. 1999. What you feel must be what you see: Adding tactile feedback to the trackpoint. In *Pro-ceedings of INTERACT '99: 7th IFIP Conference on Human–Computer Interaction*.

Cao, X., and R. Balakrishnan. 2006. Interacting with dynamically defined information spaces using a handheld projector and a pen. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*. Montreux, Switzerland: ACM.

Cao, X., C. Forlines, and R. Balakrishnan. 2007. Multi-user interaction using handheld projectors. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*. Newport, RI: ACM.

Cao, X., A. D. Wilson, R. Balakrishnan, K. Hinckley, and S. E. Hudson. 2008a. ShapeTouch: Leveraging contact shape on interactive surfaces. In *Horizontal Interactive Human Computer Systems, 2008. TABLETOP 2008*. 3rd IEEE International Workshop, 1–3. Amsterdam.

Cao, X., A. D. Wilson, R. Balakrishnan, K. Hinckley, and S. E. Hudson. 2008b. ShapeTouch: Leveraging contact shape on interactive surfaces. Paper read at IEEE TABLETOP 2008 International Workshop on Horizontal Interactive Human Computer Systems.

Card, S., W. English, and B. J. Burr. 1978. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics* 21:601–13.

Card, S., J. Mackinlay, and G. Robertson. 1990. The Design Space of Input Devices. Paper read at Proceedings of ACM CHI '90 Conference on Human Factors in Computing Systems.

Card, S., J. Mackinlay, and G. Robertson. 1991. A Morphological Analysis of the Design Space of Input Devices. *ACM Trans Inf Syst* 9(2):99–122.

Card, S., T. Moran, and A. Newell. 1980. The keystroke-level model for user performance time with interactive systems. *Commun ACM* 23(7):396–410.

Chen, N., F. Guimbretiere, M. Dixon, C. Lewis, and M. Agrawala. 2008. Navigation techniques for dual-display e-book readers. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*. Florence, Italy: ACM.

Clarkson, E., J. Clawson, K. Lyons, and T. Starner. 2005. An empirical study of typing rates on mini-QWERTY keyboards. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. Portland, OR: ACM.

Cohen, P. R., M. Dalrymple, D. B. Moran, F. C. Pereira, and J. W. Sullivan. 1989. Synergistic use of direct manipulation and natural language. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Wings for the Mind*. ACM.

Cohen, P., M. Johnston, D. McGee, S. Oviatt, J. Pittman, I. Smith, L. Chen, and J. Clow. 1997. QuickSet: Multimodal Interaction for Distributed Applications. Paper read at ACM Multimedial 97.

Collomb, M., M. Hascoët, P. Baudisch, and B. Lee. 2005. Improving drag-and-drop on wall-size displays. In *Proceedings of Graphics Interface 2005*. Victoria, BC: Canadian Human-Computer Communications Society.

Cunningham, H. A., and R. B. Welch. 1994. Multiple concurrent visual-motor mappings: Implications for models of adaptation. *J Exp Psychol Hum Percept Perform* 20(5):987–99.

Cutrell, E., D. Robbins, S. Dumais, and R. Sarin. 2006. Fast, Flexible Filtering with Phlat: Personal Search and Organization Made Easy. Paper read at CHI 2006.

Czerwinski, M., D. S. Tan, and G. G. Robertson. 2002. Women take a wider view. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems: Changing our World, Changing Ourselves*. Minneapolis, MN: ACM.

Davidson, P. L., and J. Y. Han. 2008. Extending 2D object arrangement with pressure-sensitive layering cues. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*. Monterey, CA: ACM.

Deasy, C. M., and T. E. Lasswell. 1985. *Designing Places for People: A Handbook on Human Behavior for Architects, Designers, and Facility Managers*. New York: Whitney Library of Design an imprint of Watson-Guptill Publications.

Dietz, P., and D. Leigh. 2001. DiamondTouch: A multi-user touch technology In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*. Orlando, FL: ACM Press.

Dietz, P., and W. Yerazunis. 2001. Real-Time Audio Buffering for Telephone Applications. Paper read at Proceedings of ACM UIST 2001 Symposium on User Interface Software & Technology, Orlando, FL.

Dillon, R. F., J. D. Eday, and J. W. Tombaugh. 1990. Measuring the True Cost of Command Selection: Techniques and Results. Paper read at Proceedings of ACM CHI '90 Conference on Human Factors in Computing Systems.

Douglas, S. A., A. E. Kirkpatrick, and I. S. MacKenzie. 1999. Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*. Pittsburgh, PA: ACM.

Douglas, S., and A. Mithal. 1994. The Effect of Reducing Homing Time on the Speed of a Finger-Controlled Isometric Pointing Device. Paper read at Proceedings ACM CHI '94 Conference on Human Factors in Computing Systems.

Douglas, S. A., and A. K. Mithal. 1997. *Ergonomics of Computer Pointing Devices*, *Advanced Perspectives in Applied Computing*. New York: Springer-Verlag.

Dumais, S., E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. Robbins. 2003. Stuff I've Seen: A system for personal information retrieval and re-use. Paper read at SIGIR 2003.

Elrod, S., R. Bruce, R. Gold, D. Goldberg, F. Halasz, W. Jannsen, D. Lee et al. 1992. Liveboard: A large interactive display supporting group meetings, presentations, and remote collaboration. Paper read at ACM CHI 1992 Conference on Human Factors in Computing Systems, New York.

Engelhard, L. 2008. Native Dual Mode Digitizers: Supporting Pen, Touch and Multi-Touch Inputs in One Device on any LCD. Paper read at Society for Information Display SID 08 Digest.

English, W. K., D. C. Englebart, and M. L. Berman. 1967. Display-selection Techniques for Text Manipulation. *Trans Hum Factors Electron* 8(1):5–15.

Fails, J. A., and D. R. Olsen. 2003. Interactive machine learning. Paper read at ACM Intelligent User Interfaces (IUI '03).

Fitts, P. M. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *J Exp Psychol* 47:381–91.

Fitzmaurice, G. W. 1993. Situated information spaces and spatially aware palmtop computers. *Commun ACM* 36(7):38–49.

Fitzmaurice, G. W., R. Balakrisnan, and G. Kurtenbach. 1999. Sampling, synthesis, and input devices. *Commun ACM* 42(8):54–63.

Fitzmaurice, G. W., and W. Buxton. 1994. The Chameleon: Spatially aware palmtop computers. In *Conference Companion on Human Factors in Computing Systems*. Boston, Massachusetts: ACM.

Fitzmaurice, G. W., and W. Buxton. 1997. An empirical evaluation of graspable user interfaces: Towards specialized, space-multiplexed input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Fitzmaurice, G. W., H. Ishii, and W. A. S. Buxton. 1995. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Denver, CO: ACM Press/Addison-Wesley Publishing Co.

Fitzmaurice, G. W., A. Khan, W. Buxton, G. Kurtenbach, and R. Balakrishnan. 2003. Sentient data access via a diverse society of devices. *ACM Queue* 1(8):52–68.

Fitzmaurice, G., A. Khan, R. Pieké, B. Buxton, and G. Kurtenbach. 2003. Tracking menus. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*. Vancouver, Canada: ACM.

Foley, J. D., V. L. Wallace, and P. Chan. 1984. The human factors of computer graphics interaction techniques. *IEEE Comput Graph Appl* 4(11):13–48.

Forlines, C., R. Balakrishnan, P. Beardsley, J. van Baar, and R. Raskar. 2005. Zoom-and-pick: Facilitating visual zooming and precision pointing with interactive handheld projectors. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*. Seattle, WA: ACM.

Forlines, C., and C. Shen. 2005. DTLens: Multi-user tabletop spatial data exploration. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*. Seattle, WA: ACM.

Forlines, C., C. Shen, D. Wigdor, and R. Balakrishnan. 2006. Exploring the effects of group size and display configuration on visual search. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*. Banff, AB: ACM.

Forlines, C., D. Vogel, and R. Balakrishnan. 2006. HybridPointing: Fluid switching between absolute and relative pointing with a direct input device. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*. Montreux, Switzerland: ACM.

Freeman, D., H. Benko, M. R. Morris, and D. Wigdor. 2009. ShadowGuides: Visualizations for in-situ learning of multi-touch and whole-hand gestures. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. Banff, AB: ACM.

Freeman, W., and C. Weissman. 1995. Television control by hand gestures. Paper read at International Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland.

Frisch, M., J. Heydekorn, and R. Dachselt. 2009. Investigating Multi-Touch and Pen Gestures for Diagram Editing on Interactive Surfaces. Paper read at ITS '09 Conference on Interactive Tabletops and Surfaces.

Funkhouser, T., and K. Li. 2000. Onto the Wall: Large Displays. *IEEE Comput Graph Appl (Special issue)* 20(4).

Gibson, J. J. 1962. Observations on active touch. *Psychol Rev* 69:477–91.

Goldberg, D., and C. Richardson. 1993. Touch-typing with a stylus. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*. Amsterdam, The Netherlands: ACM.

Grossman, T., and R. Balakrishnan. 2005. The bubble cursor: Enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Grossman, T., K. Hinckley, P. Baudisch, M. Agrawala, and R. Balakrishnan. 2006. Hover widgets: Using the tracking state to extend the capabilities of pen-operated devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montréal, QC: ACM.

Grossman, T., D. Wigdor, and R. Balakrishnan. 2004. Multi-finger gestural interaction with 3d volumetric displays. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. Santa Fe, NM: ACM.

Grudin, J. 2001. Partitioning digital worlds: Focal and peripheral awareness in multiple monitor use. Paper read at CHI 2001.

Guiard, Y. 1987. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *J Motor Behav* 19(4):486–517.

Guiard, Y., F. Buourgeois, D. Mottet, and M. Beaudouin-Lafon. 2001. Beyond the 10-bit Barrier: Fitts' Law in Multi-Scale Electronic Worlds. Paper read at IHM-HCI 2001, Sept., Lille, France.

Guimbretiere, F., A. Martin, and T. Winograd. 2005. Benefits of merging command selection and direct manipulation. *ACM Trans Comput Hum Interact* 12(3):460–76.

Gunawardana, A., T. Paek, and C. Meek. 2010. Usability guided key-target resizing for soft keyboards. In *Proceeding of the 14th International Conference on Intelligent User Interfaces*. Hong Kong, China: ACM.

Hancock, M., and S. Carpendale. 2007. Supporting multiple Off-Axis viewpoints at a tabletop display. Paper read at Horizontal Interactive Human-Computer Systems, 2007. TABLETOP '07. Second Annual IEEE International Workshop on, 10–12 Oct. 2007.

Hancock, M. S., F. D. Vernier, D. Wigdor, S. Carpendale, and S. Chia. 2006. Rotation and translation mechanisms for tabletop interaction. Paper read at Horizontal Interactive Human-Computer Systems, 2006. TableTop 2006. First IEEE International Workshop on, 5–7 Jan. 2006.

Harrison, B. L., K. P. Fishkin, A. Gujar, C. Mochon, and R. Want. 1998. Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Los Angeles, CA: ACM Press/Addison-Wesley Publishing Co.

Harrison, B., H. Ishii, K. Vicente, and W. Buxton. 1995. Transparent layered user interfaces: An evaluation of a display design to enhance focused and divided attention. Paper read at Proceedings of CHI '95: ACM Conference on Human Factors in Computing Systems.

Hartmann, B., M. Ringel Morris, H. Benko, and A. D. Wilson. 2009. Augmenting interactive tables with mice and keyboards. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Hauptmann, A. 1989. Speech and gestures for graphic image manipulation. Paper read at Proceedings of CHI '89: ACM Conference on Human Factors in Computing Systems, Apr. 30–May 4, Austin, TX.

Hilliges, O., S. Izadi, A. D. Wilson, S. Hodges, A. Garcia-Mendoza, and A. Butz. 2009. Interactions in the air: Adding further depth to interactive tabletops. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Hinckley, K., P. Baudisch, G. Ramos, and F. Guimbretiere. 2005. Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Hinckley, K., E. Cutrell, S. Bathiche, and T. Muss. 2002. Quantitative analysis of scrolling techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*. Minneapolis, MN: ACM.

Hinckley, K., M. Czerwinski, and M. Sinclair. 1998. Interaction and modeling techniques for desktop two-handed input. Paper read at ACM UIST 1998 Symposium on User Interface Software and Technology, New York.

Hinckley, K., M. Dixon, R. Sarin, F. Guimbretiere, and R. Balakrishnan. 2009. Codex: A dual screen tablet computer. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*. Boston, MA: ACM.

Hinckley, K., F. Guimbretiere, P. Baudisch, R. Sarin, M. Agrawala, and E. Cutrell. 2006. The springboard: Multiple modes in one spring-loaded control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montréal, QC: ACM.

Hinckley, K., M. Pahud, and B. Buxton. 2010. Direct display interaction via Simultaneous Pen + Multi-touch Input. Paper read at Society for Information Display SID 2010 Digest.

Hinckley, K., R. Pausch, J. C. Goble, and N. F. Kassell. 1994a. A survey of design issues in spatial input. Paper read at ACM UIST 1994 Symposium on User Interface Software and Technology, New York.

Hinckley, K., R. Pausch, J. C. Goble, and N. F. Kassell. 1994b. A three-dimensional user interface for neurosurgical visualization. Paper read at Proceedings of SPIE Vol. 2164, Medical Imaging 1994: Image Capture, Formatting, and Display.

Hinckley, K., R. Pausch, D. Proffitt, and N. E. Kassell. 1998. Two-handed virtual manipulation. *ACM Trans Comput Hum Interact* 5(3):260–302.

Hinckley, K., J. Pierce, E. Horvitz, and M. Sinclair. 2005. Foreground and Background Interaction with Sensor-Enhanced Mobile Devices. *ACM TOCHI* 12(1 [Special Issue on Sensor-Based Interaction]):31–52.

Hinckley, K., J. Pierce, M. Sinclair, and E. Horvitz. 2000. Sensing Techniques for Mobile Interaction. Paper read at ACM UIST 2000 Symposium on User Interface Software & Technology, San Diego, CA.

Hinckley, K., G. Ramos, F. Guimbretiere, P. Baudisch, and M. Smith. 2004. Stitching: Pen Gestures that Span Multiple Displays. Paper read at ACM 7th International Working Conference on Advanced Visual Interfaces (AVI 2004), May 25–28, Gallipoli (Leece), Italy.

Hinckley, K., and M. Sinclair. 1999. Touch-Sensing Input Devices. Paper read at ACM CHI '99 Conference on Human Factors in Computing Systems.

Hinckley, K., and S. Hyunyoung. 2011. Sensor synaesthesia: Touch in motion, and motion in touch. In *Proceedings of the 2011 annual conference on Human factors in computing systems* (CHI '11), 801–10. New York: ACM.

Hinckley, K., K. Yatani, M. Pahud, N. Coddington, J. Rodenhouse, A. Wilson, H. Benko, and B. Buxton. 2010. Pen + touch = new tools. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. New York: ACM.

Hinckley, K., S. Zhao, R. Sarin, P. Baudisch, E. Cutrell, M. Shilman, and D. Tan. 2007. InkSeine: In Situ Search for Active Note Taking. Paper read at CHI 2007.

Hinrichs, U., M. Hancock, C. Collins, and S. Carpendale. 2007. Examination of Text-Entry methods for tabletop displays. Paper read at Horizontal Interactive Human-Computer Systems, 2007. TABLETOP '07. Second Annual IEEE International Workshop on, 10–12 Oct. 2007.

Hirshfield, L. M., K. Chauncey, R. Gulotta, A. Girouard, E. T. Solovey, R. J. Jacob, A. Sassaroli, and S. Fantini. 2009. Combining Electroencephalograph and Functional Near Infrared Spectroscopy to Explore Users' Mental Workload. In *Proceedings of the 5th International Conference on Foundations of Augmented Cognition. Neuroergonomics and Operational Neuroscience: Held as Part of HCI International 2009*. San Diego, CA: Springer-Verlag.

Hirshfield, L. M., E. T. Solovey, A. Girouard, J. Kebinger, R. J. K. Jacob, A. Sassaroli, and S. Fantini. 2009. Brain measurement for usability testing and adaptive interfaces: An example of uncovering syntactic workload with functional near infrared spectroscopy. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*. Boston, MA: ACM.

Holman, D., and R. Vertegaal. 2008. Organic user interfaces: Designing computers in any way, shape, or form. *Commun ACM* 51(6):48–55.

Holz, C., and P. Baudisch. 2010. The generalized perceived input point model and how to double touch accuracy by extracting fingerprints. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Honan, M., E. Serina, R. Tal, and D. Rempel. 1995. Wrist Postures While Typing on a Standard and Split Keyboard. In *Proceedings of HFES Human Factors and Ergonomics Society 39th Annual Meeting* 39(5):366–8.

Horvitz, E. 1999. Principles of Mixed-Initiative user interfaces. Paper read at Proceedings of ACM CHI '99 Conference on Human Factors in Computing Systems, Pittsburgh, PA.

Horvitz, E., J. Breese, D. Heckerman, D. Hovel, and K. Rommelse. 1998. The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. Paper read at Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, July 1998, 256–65. San Francisco, CA: Morgan Kaufmann.

Horvitz, E., A. Jacobs, and D. Hovel. 1999. Attention-sensitive alerting. Paper read at Proceedings of UAI '99, Conference on Uncertainty and Artificial Intelligence, July, Stockholm, Sweden.

Hutchings, D. R., and J. Stasko. 2004. Shrinking window operations for expanding display space. In *Proceedings of the Working Conference on Advanced Visual Interfaces*. Gallipoli, Italy: ACM.

Hutchings, D. R., and J. Stasko. 2005. Mudibo: Multiple dialog boxes for multiple monitors. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. Portland, OR: ACM.

Ishii, H., and B. Ullmer. 1997. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Jacob, R. J. K., A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum. 2008. Reality-based interaction: A framework for post-WIMP interfaces. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*. Florence, Italy: ACM.

Jacob, R. J. K., L. E. Sibert, D. C. McFarlane, and M. Preston Mullen Jr. 1994. Integrality and separability of input devices. *ACM Trans Comput Hum Interact* 1(1):3–26.

Jellinek, H. D., and S. K. Card. 1990. Powermice and user performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People*. Seattle, WA: ACM.

John, B. E., and D. Kieras. 1996. Using GOMS for User Interface Design and Evaluation: Which Technique? *ACM Trans Comput Hum Interact* 3(4):287–319.

Jojic, N., B. Brumitt, B. Meyers, and S. Harris. 2000. Detecting and estimating pointing gestures in dense disparity maps. Paper read at Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition.

Kabbash, P., and W. A. S. Buxton. 1995. The "prince"; technique: Fitts' law and selection using area cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Denver, CO: ACM Press/Addison-Wesley Publishing Co.

Kabbash, P., W. Buxton, and A. Sellen. 1994. Two-handed input in a compound task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Celebrating Interdependence*. Boston, MA: ACM.

Kamba, T., S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. 1996. Using small screen space more efficiently. Paper read at Conference Proceedings on Human Factors in Computing Systems.

Karat, C. -M., C. Halverson, D. Horn, and J. Karat. 1999. Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*. Pittsburgh, PA: ACM.

Karl, L., M. Pettey, and B. Shneiderman. 1993. Speech-Activated versus Mouse-Activated Commands for Word Processing Applications: An Empirical Evaluation. *Int J Man Mach Stud* 39(4):667–87.

Karlson, A. K., and B. B. Bederson. 2007. ThumbSpace: Generalized one-handed input for touchscreen-based mobile devices. In *Proceedings of Interact*, Springer *Verlag*.

Karlson, A. K., B. B. Bederson, and J. SanGiovanni. 2005. AppLens and launchTile: two designs for one-handed thumb use on small devices. In *Proceedings of CHI 2005*, Association for Computing Machinery, Inc.

Khan, A., J. Matejka, G. Fitzmaurice, and G. Kurtenbach. 2005. Spotlight: Directing users' attention on large displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Kirsh, D. 1995. The intelligent use of space. *Artif Intell* 73:31–68.

Kirsh, D., and P. Maglio. 1994. On distinguishing epistemic from pragmatic action. *Cogn Sci* 18(4):513–49.

Klemmer, S. R., M. W. Newman, R. Farrell, M. Bilezikjian, and J. A. Landay. 2001. The designers' outpost: A tangible interface for collaborative web site. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*. Orlando, FL: ACM.

Ko, B. K., and H. S. Yang. 1997. Finger mouse and gesture recognition system as a new human computer interface. *Comput Graph* 21(5):555–61.

Kristensson, P.-O., and S. Zhai. 2004. SHARK 2: A large vocabulary shorthand writing system for pen-based computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. Santa Fe, NM: ACM.

Krueger, M. 1991. VIDEOPLACE and the interface of the future. In *The Art of Human Computer Interface Design*, ed. B. Laurel, 417–22. Menlo Park, CA: Addison-Wesley.

Krueger, M., T. Gionfriddo, and K. Hinrichsen. 1985. VIDEOPLACE — An artificial reality. Paper read at Proceedings of CHI '85: ACM Conference on Human Factors in Computing Systems, Apr. 14–18, San Francisco, CA.

Krumm, J., and K. Hinckley. 2004. The NearMe wireless proximity Server. Paper read at Ubicomp 2004.

Kurtenbach, G., and B. Buxton. 1991a. GEdit: A test bed for editing by contiguous gestures. *SIGCHI Bull* 23(2):22–6.

Kurtenbach, G., and W. Buxton. 1991b. Issues in combining marking and direct manipulation techniques. In *Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology*. Hilton Head, SC: ACM.

Kurtenbach, G., G. Fitzmaurice, T. Baudel, and B. Buxton. 1997. The design of a GUI paradigm based on tablets, two-hands, and transparency. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Kurtenbach, G., A. Sellen, and W. Buxton. 1993. An emprical evaluation of some articulatory and cognitive aspects of 'marking menus.' *J Hum Comput Interact* 8(1):1–23.

Lank, E., and E. Saund. 2005. Sloppy Selection: Providing an Accurate Interpretation of Imprecise Selection Gestures. *Comput Graph* 29(4):490–500.

Lansdale, M., and E. Edmonds. 1992. Using memory for events in the design of personal filing systems. *Int J Man Mach Stud* 36(1):97–126.

LaViola, J. and R. Zeleznik. 2004. MathPad2: A System for the Creation and Exploration of Mathematical Sketches. *ACM Trans Graph* 23(3):432–40.

Lepinski, G. J., T. Grossman, and G. Fitzmaurice. 2010. The design and evaluation of multitouch marking menus. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Levine, S., and S. Ehrlich. 1991. The Freestyle system: A design perspective. In *Human-Machine Interactive Systems*, ed. A. Klinger. New York: Plenum Press.

Lewis, J., K. Potosnak, and R. Magyar. 1997. Keys and Keyboards. In *Handbook of Human–Computer Interaction*, ed. M. Helander, T. Landauer and P. Prabhu. Amsterdam: North-Holland.

Li, Y., K. Hinckley, Z. Guan, and J. A. Landay. 2005. Experimental analysis of mode switching techniques in pen-based user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Lipscomb, J., and M. Pique. 1993. Analog Input Device Physical Characteristics. *SIGCHI Bull* 25(3):40–5.

Lyons, K., C. Skeels, T. Starner, C. M. Snoeck, B. A. Wong, and D. Ashbrook. 2004. Augmenting conversations using Dual-Purpose speech. Paper read at UIST'04 Symposium on User Interface Software and Technology, Sante Fe, NM.

Mackay, W. E. 2002. Which Interaction Technique Works When? Floating Palettes, Marking Menus and Toolglasses Support Different Task Strategies. Paper read at Proceedings of AVI 2002 International Conference on Advanced Visual Interfaces.

MacKenzie, I. S. 1989. A note on the information-theoretic basis for Fitts' Law. *J Motor Behav* 21:323–30.

MacKenzie, I. S. 1992a. Fitts' law as a research and design tool in human–computer interaction. *Hum Comput Interact* 7:91–139.

MacKenzie, I. S. 1992b. Movement time prediction in human–computer interfaces. In *Proceedings of the Conference on Graphics Interface '92*. Vancouver, BC: Morgan Kaufmann Publishers Inc.

MacKenzie, I. S. 1995. Input Devices and Interaction Techniques for Advanced Computing. In *Virtual Environments and Advanced Interface Design*, ed. W. Barfield and T. Furness, 437–70. Oxford, UK: Oxford University Press.

MacKenzie, I. S., and Y. Guiard. 2001. The two-handed desktop interface: Are we there yet? In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. Seattle, WA: ACM.

MacKenzie, I. S., H. Kober, D. Smith, T. Jones, and E. Skepner. 2001. LetterWise: Prefix-based disambiguation for mobile text input. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*. Orlando, FL: ACM.

MacKenzie, I. S., and A. Oniszczak. 1998. A comparison of three selection techniques for touchpads. Paper read at Proceedings of ACM CHI '98 Conference on Human Factors in Computing Systems.

MacKenzie, I. S., A. Sellen, and W. Buxton. 1991. A comparison of input devices in elemental pointing and dragging tasks. Paper read at Proceedings of ACM CHI '91 Conference on Human Factors in Computing Systems.

MacKenzie, I. S., and R. W. Soukoreff. 2002. A model of two-thumb text entry. In *Proceedings of Graphics Interface*. Toronto: Canadian Information Processing Society.

MacKenzie, I. S., and C. Ware. 1993. Lag as a determinant of human performance in interactive systems. Paper read at Proceedings of ACM INTERCHI '93 Conference on Human Factors in Computing Systems.

MacKenzie, I. S., and S. X. Zhang. 1997. The immediate usability of graffiti. In *Proceedings of the conference on Graphics interface '97*, eds. W. A. Davis, M. Mantei, and R. V. Klassen, 129–37. Toronto, Canada: Canadian Information Processing Society.

Mackinlay, J. D., and J. Heer. 2004. Wideband displays: Mitigating multiple monitor seams. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*. Vienna, Austria: ACM.

Maes, P., T. Darrell, B. Blumberg, and A. Pentland. 1997. The ALIVE system: Wireless, full-body interaction with autonomous agents. *ACM Multimedia Syst* (Special Issue on Multimedia and Multisensory Virutal Worlds) 5:105–12.

Malik, S., A. Ranjan, and R. Balakrishnan. 2005. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*. Seattle, WA: ACM.

Marklin, R., G. Simoneau, and J. Monroe. 1997. The Effect of Split and Vertically-Inclined Computer Keyboards on Wrist and Forearm Posture. In *Proceedings HFES Human Factors and Ergonomics Society 41st Annual Meeting* 41(13):1071–813.

Matsushita, M., M. Iida, T. Ohguro, Y. Shirai, Y. Kakehi, and T. Naemura. 2004. Lumisight table: A face-to-face collaboration support system that optimizes direction of projected information to each stakeholder In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*. Chicago, IL: ACM Press.

Matsushita, N., and J. Rekimoto. 1997. HoloWall: Designing a finger, hand, body, and object sensitive wall. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*. Banff, AB: ACM.

McCallum, D. C., and P. Irani. 2009. ARC-Pad: Absolute + relative cursor positioning for large displays with a mobile touchscreen. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

McLoone, H., K. Hinckley, and E. Cutrell. 2003. Bimanual Interaction on the Microsoft Office Keyboard. In *INTERACT 2003*.

Microsoft Corp. 2002. Windows XP pointer ballistics. http://msdn.microsoft.com/en-us/windows/hardware/gg463319.aspx (accessed on Oct 3, 2011).

Microsoft Courier. 2010. In *Wikipedia the Free Encyclopedia*. http://en.wikipedia.org/wiki/Microsoft_Courier (accessed on Oct 3, 2011).

Mine, M., F. Brooks, and C. Sequin. 1997. Moving objects in space: Exploiting proprioception in virtual environment interaction. In *ACM SIGGRAPH 1997 Conference on Computer Graphics and Interactive Techniques*. New York: ACM.

Moran, T. P., P. Chiu, and W. van Melle. 1997. Pen-based interaction techniques for organizing material on an electronic whiteboard. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*. Banff, AB: ACM.

Morris, M. R., A. J. B. Brush, and B. R. Meyers. 2008. A field study of knowledge workers'; use of interactive horizontal displays. Paper read at Horizontal Interactive Human Computer Systems, 2008. *TABLETOP 2008*. 3rd IEEE International Workshop on, 1–3 Oct. 2008.

Morris, M. R., A. Paepcke, T. Winograd, and J. Stamberger. 2006. TeamTag: Exploring centralized versus replicated controls for co-located tabletop groupware. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Montrëal, QC: ACM.

Moscovich, T. 2009. Contact area interaction with sliding widgets. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Moscovich, T., and J. F. Hughes. 2006. Multi-finger cursor techniques. In *Proceedings of Graphics Interface 2006*. Quebec, Canada: Canadian Information Processing Society.

Myers, B., H. Stiel, and R. Gargiulo. 1998. Collaboration using multiple PDAs connected to a PC. Paper read at Proceedings of ACM CSCW '98 Conference on Computer Supported Cooperative Work, Nov. 14–18, Seattle, WA.

Myers, B. A., K. P. Lie, and B. -C. Yang. 2000. Two-handed input using a PDA and a mouse. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. The Hague, The Netherlands: ACM.

Nacenta, M. A., D. Aliakseyeu, S. Subramanian, and C. Gutwin. 2005. A comparison of techniques for multi-display reaching. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Portland, OR: ACM.

Ni, T., and P. Baudisch. 2009. Disappearing mobile devices. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Nickel, K., and R. Stiefelhagen. 2003. Pointing gesture recognition based on 3D-tracking of face, hands and head orientation. In *Proceedings of the 5th International Conference on Multimodal Interfaces*. Vancouver, BC: ACM.

Norman, D. 1990. *The Design of Everyday Things*. New York: Doubleday.

Oakley, I., S. Brewster, and P. Gray. 2001. Solving multi-target haptic problems in menu interaction. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. Seattle, WA: ACM.

O'Hara, K., and A. Sellen. 1997. A comparison of reading paper and on-line documents. Paper read at CHI '97.

Olson, J. R., and G. M. Olson. 1990. The growth of cognitive modeling in human–computer interaction since GOMS. *Hum Comput Interact* 5(2):221–65.

Oviatt, S., A. DeAngeli, and K. Kuhn. 1997. Integration and synchronization of input modes during multimodal human–computer interaction. In *Referring Phenomena in a Multimedia Context and their Computational Treatment*. Madrid, Spain: Association for Computational Linguistics.

Pederson, E., K. McCall, T. Moran, and F. Halasz. 1993. Tivoli: An electronic whiteboard for informal workgroup meetings. Paper read at ACM CHI Conference on Human Factors in Computing Systems, New York.

Potter, R. L., L. J. Weldon, and B. Shneiderman. 1988. Improving the accuracy of touch screens: An experimental evaluation of three strategies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Washington, DC: ACM.

Poupyrev, I., and S. Maruyama. 2003. Tactile interfaces for small touch screens. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*. Vancouver, Canada: ACM.

Price, M. N., B. Schilit, and G. Golovchinsky. 1998. XLibris: The active reading machine. Paper read at CHI'98 Extended Abstracts.

Putz-Anderson, V. 1988. *Cumulative Trauma Disorders: A Manual for Musculoskeletal Diseases of the Upper Limbs*. Bristol, PA: Taylor & Francis.

Ramos, G., and R. Balakrishnan. 2005. Zliding: Fluid zooming and sliding for high precision parameter manipulation. Paper read at UIST 2006.

Ramos, G., and R. Balakrishnan. 2006. Pressure marks. Paper read at UNPUBLISHED MANUSCRIPT (under review).

Ramos, G., M. Boulos, and R. Balakrishnan. 2004. Pressure widgets. Paper read at CHI 2004.

Raskar, R., P. Beardsley, J. van Baar, Y. Wang, P. Dietz, J. Lee, D. Leigh, and T. Willwacher. 2004. RFIG lamps: Interacting with a self-describing world via photosensing wireless tags and projectors. 23(3):406–15.

Raskar, R., J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. 2005. iLamps: Geometrically aware and self-configuring projectors. In *ACM SIGGRAPH 2005 Courses*. Los Angeles, CA: ACM.

Raskin, J. 2000. *The Humane Interface: New Directions for Designing Interactive Systems.* Boston, MA: Addison-Wesley.

Rekimoto, J. 1996. Tilting operations for small screen interfaces. Paper read at ACM UIST Symposium on User Interface Software and Technology, New York.

Rekimoto, J. 1998. A multiple-device approach for supporting whiteboard based interaction. Paper read at ACM CHI Conference on Human Factors in Computing Systems, New York.

Rekimoto, J., Y. Ayatsuka, M. Kohno, and H. Oba. 2003. Proximal interactions: A direct manipulation technique for wireless networking. Paper read at INTERACT 2003.

Rempel, D., J. Bach, L. Gordon, and R. Tal. 1998. Effects of forearm pronation/supination on carpal tunnel pressure. *J Hand Surg* 23(1):38–42.

Rime, B., and L. Schiaratura. 1991. Gesture and speech. In *Fundamentals of Nonverbal Behaviour*. New York: Press Syndacate of the University of Cambridge.

Rutledge, J. D., and T. Selker. 1990. Force-to-motion functions for pointing. In *Proceedings of the IFIP TC13 Third Interational Conference on Human–Computer Interaction*, 701–6. Amsterdam: North-Holland Publishing Co.

Ryall, K., C. Forlines, C. Shen, and M. R. Morris. 2004. Exploring the effects of group size and table size on interactions with tabletop shared-display groupware. In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*. Chicago, IL: ACM.

Saponas, T. S., D. S. Tan, D. Morris, R. Balakrishnan, J. Turner, and J. A. Landay. 2009. Enabling always-available input with muscle–computer interfaces. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Saund, E., and E. Lank. 2003. Stylus input and editing without prior selection of mode. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*. Vancouver, Canada: ACM.

Sawhney, N., and C. M. Schmandt. 2000. Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments. *ACM Trans Comput Hum Interact* 7(3):353–83.

Schilit, B. N., N. I. Adams, and R. Want. 1994. Context-Aware computing applications. Paper read at Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, Dec., Santa Cruz, CA.

Schmandt, C. M., N. Marmasse, S. Marti, N. Sawhney, and S. Wheeler. 2000. Everywhere messaging. *IBM Syst J* 39(3 & 4).

Schmidt, A., M. Beigl, and H. W. Gellersen. 1999. There is more to context than location. *Comput Graph* 23(6):893–901.

Schwarz, J., S. Hudson, J. Mankoff, and A. D. Wilson. 2010. A framework for robust and flexible handling of inputs with uncertainty. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. New York: ACM.

Scott, S. D., M. Sheelagh, T. Carpendale, and K. M. Inkpen. 2004. Territoriality in collaborative tabletop workspaces. In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*. Chicago, IL: ACM.

Sears, A. 1991. Improving touchscreen keyboards: Design issues and a comparison with other devices. *Interact Comput* 3(3):253–69.

Sears, A. 1993. Investigating touchscreen typing: The effect of keyboard size on typing speed. *Behav Inf Technol* 12(1):17–22.

Sears, A., and B. Shneiderman. 1991. High Precision Touchscreens: Design Strategies and Comparisons with a Mouse. *Int J Man Mach Stud* 34(4):593–613.

Sellen, A., G. Kurtenbach, and W. Buxton. 1992. The prevention of mode errors through sensory feedback. *Hum Comput Interact* 7(2):141–64.

Seow, S. C. 2008. *Designing and Engineering Time*. 1st ed. Boston, MA: Addison-Wesley.

Shell, J. S., R. Vertegaal, D. Cheng, A. W. Skaburskis, C. Sohn, A. J. Stewart, O. Aoudeh, and C. Dickie. 2004. ECSGlasses and EyePliances: Using attention to open sociable windows of interaction. In *Proceedings of the 2004 Symposium on Eye Tracking Research and Applications*. San Antonio, TX: ACM.

Shen, C., K. Everitt, and K. Ryall. 2003. UbiTable: Impromptu Face-to-Face collaboration on horizontal interactive surfaces. Paper read at UbiComp 2003.

Shen, C., F. D. Vernier, C. Forlines, and M. Ringel. 2004. DiamondSpin: An extensible toolkit for around-the-table interaction. In *Proceedings of the 2004 Conference on Human Factors in Computing Systems*. Vienna, Austria: ACM Press.

Smutz, W., E. Serina, T. Bloom, and D. Rempel. 1994. A System for Evaluating the Effect of Keyboard Design on Force, Posture, Comfort, and Productivity. *Ergonomics* 37(10):1649–60.

Snibbe, S., and K. MacLean. 2001. Haptic techniques for media control. *CHI Lett (Proc UIST 2001)* 3(2):199–208.

Sommer, R. 1965. Further studies of small group ecology. *Sociometry* 28:337–48.

Song, H., T. Grossman, G. Fitzmaurice, F. Guimbretiere, A. Khan, R. Attar, and G. Kurtenbach. 2009. PenLight: Combining a mobile projector and a digital pen for dynamic visual overlay. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*. Boston, MA: ACM.

Song, H., F. Guimbretiere, T. Grossman, and G. Fitzmaurice. 2010. MouseLight: Bimanual interactions on digital paper using a pen and a spatially-aware mobile projector. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*. Atlanta, GA: ACM.

Stifelman, L. J. 1996. Augmenting real-world objects: A paper-based audio notebook. In *Conference Companion on Human Factors in Computing Systems: Common Ground*. Vancouver, BC: ACM.

Streitz, N. A., J. Geuler, T. Holmer, S. Konomi, C. Miller-Tomfelde, W. Reischl, P. Rexroth, P. Seitz, and R. Steinmetz. 1999. i-LAND: An interactive landscape for creativity and innovation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*. Pittsburgh, PA: ACM Press.

Surati, R. J. 1999. Scalable self-calibrating display technology for seamless large-scale displays. Massachusetts Institute of Technology.

Sutherland, I. E. 1964. Sketch pad a man-machine graphical communication system. In *Proceedings of the SHARE Design Automation Workshop*. ACM. See http://doi.acm.org/10.1145/800265.810742.

Swaminathan, K., and S. Sato. 1997. Interaction design for large displays. *Interactions* 4(1):15–24.

Tan, D. S., and M. Czerwinski. 2003. Effects of Visual Separation and Physical Discontinuities when Distributing Information across Multiple Displays. In *OZCHI 2003 Conference for the Computer-Human Interaction Special Interest Group of the Ergonomics Society of Australia*.

Tan, D. S., D. Gergle, P. Scupelli, and R. Pausch. 2003. With similar visual angles, larger displays improve spatial performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Ft. Lauderdale, FL: ACM.

Tang, A., M. Pahud, K. Inkpen, H. Benko, J. C. Tang, and B. Buxton. 2010. Three's company: Understanding communication channels in three-way distributed collaboration. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*. Savannah, GA: ACM.

Tilbrook, D. M. 1976. *A Newspaper Pagination System*. Toronto, ON: Department of Computer Science, University of Toronto.

Toney, A., and B. H. Thomas. 2006. Considering reach in tangible and table top design. In *Horizontal Interactive Human-Computer Systems, International Workshop On*, 57–8. First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06).

Trevor, J., D. M. Hilbert, B. N. Schilit, and T. K. Koh. 2001. From desktop to phonetop: a UI for web interaction on very small devices. In *Proceedings of the 14th annual ACM symposium on User interface software and technology* (UIST '01), 121–130. New York, NY, USA: ACM.

Trimble, J., R. Wales, and R. Gossweiler. 2003. NASA's MERBoard. In *Public and Situated Displays: Social and Interactional Aspects of Shared Display Technologies*, ed. K. O'Hara, M. Perry, E. Churchill and D. Russell. Kluwer.

Tse, E., S. Greenberg, C. Shen, C. Forlines, and R. Kodama. 2008. Exploring true multi-user multimodal interaction over a digital table. In *Proceedings of the 7th ACM Conference on Designing Interactive Systems*. Cape Town, SA: ACM.

Ullmer, B., and H. Ishii. 1997. The metaDESK: Models and prototypes for tangible user interfaces. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*. Banff, AB: ACM.

Underkoffler, J., and H. Ishii. 1998. Illuminating light: An optical design tool with a luminous-tangible interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Los Angeles, CA: ACM Press/Addison-Wesley Publishing Co.

Villar, N., S. Izadi, D. Rosenfeld, H. Benko, J. Helmes, J. Westhues, S. Hodges et al. 2009. Mouse 2.0: Multi-touch meets the mouse. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*. Victoria, BC: ACM.

Viviani, P., and T. Flash. 1995. Minimum-jerk, two-thirds power law and isochrony: Converging approaches to the study of movement planning. *J Exp Psychol Percept Perform* 21:32–53.

Vogel, D., and R. Balakrishnan. 2005. Distant freehand pointing and clicking on very large, high resolution displays. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*. Seattle, WA: ACM.

Vogel, D., and P. Baudisch. 2007. Shift: A technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. San Jose, CA: ACM.

Want, R., and G. Borriello. 2000. Survey on Information Appliances. *IEEE Pers Commun* 24–31.

Weiser, M. 1991. The Computer for the 21st Century. *Sci Am* 265(3): 94–104.

Wickens, C. 1992. *Engineering Psychology and Human Performance*. Upper Saddle River, NJ: Prentice Hall.

Wigdor, D., and R. Balakrishnan. 2003. TiltText: Using tilt for text input to mobile phones. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*. Vancouver, Canada: ACM.

Wigdor, D., C. Forlines, P. Baudisch, J. Barnwell, and C. Shen. 2007. Lucid touch: A See-Through mobile device. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*. Newport, RI: ACM.

Wigdor, D., C. Shen, C. Forlines, and R. Balakrishnan. 2007. Perception of elementary graphical elements in tabletop and multi-surface environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. San Jose, CA: ACM.

Wigdor, D., S. Williams, M. Cronin, R. Levy, K. White, M. Mazeev, and H. Benko. 2009. Ripples: utilizing per-contact visualizations to improve user interaction with touch displays. In Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09). ACM, New York, NY, USA, 3–12.

Wilson, F. R. 1998. *The Hand: How Its Use Shapes the Brain, Language, and Human Culture*. New York: Pantheon Books.

Wilson, A. 2005. PlayAnywhere: A Compact Interactive Tabletop Projection-Vision System.

Wilson, A. D. 2007. Depth-Sensing Video Cameras for 3D Tangible Tabletop Interaction. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '07)* 201–4.

Wilson, A. D. 2009. Simulating grasping behavior on an imaging interactive surface. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. Banff, AB: ACM.

Wilson, A. D., and H. Benko. 2010. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. New York: ACM.

Wilson, A. D., S. Izadi, O. Hilliges, A. Garcia-Mendoza, and D. Kirk. 2008. Bringing physics to the surface. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*. Monterey, CA: ACM.

Wilson, A., and S. Shafer. 2003. XWand: UI for intelligent spaces. Paper read at ACM CHI 2003 Conference on Human Factors in Computing Systems, New York.

Wilson, A., and N. Oliver. 2003. GWindows: Towards robust Perception-Based UI. Paper read at Proceedings of CVPR 2003 (Workshop on Computer Vision for HCI).

Wobbrock, J. O., M. R. Morris, and A. D. Wilson. 2009. User-defined gestures for surface computing. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*. Boston, MA: ACM.

Wu, M., and R. Balakrishnan. 2003. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. Paper read at UIST 2003, Vancouver, BC, Canada.

Yamada, H. 1980. A historical study of typewriters and typing methods: From the position of planning Japanese parallels. *J Inf Process* 24(4):175–202.

Yee, K.-P. 2003. Peephole displays: Pen interaction on spatially aware handheld computers. Paper read at CHI 2003.

Yee, K.-P., K. Swearingen, K. Li, and M. Hearst. 2003. Faceted metadata for image search and browsing. Paper read at CHI 2003.

Zeleznik, R., A. Bragdon, F. Adeputra, and H.-S. Ko. 2010. Hands-on math: A page-based multi-touch and pen desktop for technical work and problem solving. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*. New York: ACM.

Zeleznik, R., A. Bragdon, H.-S. Ko, and F. Adeputra. 2010. Hands-On Math: A page-based multi-touch and pen desktop for technical work and problem solving. Paper read at UIST 2010 Symposium on User Interface Software and Technology.

Zhai, S. 1993. Human performance evaluation of manipulation schemes in virtual environments. Paper read at Proceedings of IEEE Virtual Reality International Symposium (VRAIS '93), Seattle.

Zhai, S. 1998. User performance in relation to 3D input device design. *SIGGRAPH Comput Graph* 32(4):50–54.

Zhai, S., W. Buxton, and P. Milgram. 1996. The Partial Occlusion Effect: Utilizing Semi-transparency for Human–Computer Interaction. *ACM Trans Comput Hum Interact* 3(3):254–84.

Zhai, S., and P. Kristensson. 2003. Shorthand writing on stylus keyboard. Paper read at CHI '03 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems April 05–10, Ft. Lauderdale, FL.

Zhai, S., P. Kristensson, P. Gong, M. Greiner, S. A. Peng, L. M. Liu, and A. Dunnigan. 2009. Shapewriter on the iphone: From the laboratory to the real world. Paper read at CHI '09 Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems, April 04-09, Boston, MA.

Zhai, S., and P. Milgram. 1993. Human performance evaluation of isometric and elastic rate controllers in a 6DoF tracking task. Paper read at Proceedings of SPIE Telemanipulator Technology.

Zhai, S., P. Milgram, and W. Buxton. 1996. The influence of muscle groups on performance of multiple degree-of-freedom input. Paper read at ACM CHI 1996 Conference on Human Factors in Computing Systems, New York.

Zhai, S., B. A. Smith, and T. Selker. 1997. Improving Browsing Performance: A study of four input devices for scrolling and pointing tasks. In *Proceedings of the IFIP TC13 Interantional Conference on Human-Computer Interaction* (INTERACT '97), eds. S. Howard, J. Hammond, and G. Lindgaard, 286–93. London: Chapman & Hall, Ltd.

Zhao, S., and R. Balakrishnan. 2004. Simple vs. compound mark hierarchical marking menus. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. Santa Fe, NM: ACM.

7 Sensor- and Recognition-Based Input for Interaction

Andrew D. Wilson

CONTENTS

7.1 INTRODUCTION

Sensors convert a physical signal into an electrical signal that may be manipulated symbolically on a computer. A wide variety of sensors have been developed for aerospace, automotive, and robotics applications (Fraden 2003). Continual innovations in manufacturing and reductions in cost have allowed many sensing technologies to find application in consumer products. An interesting example is the development of the ubiquitous computer mouse. Douglas Engelbart's original mouse, so named because its wire "tail" came out of its end, used two metal wheels and a pair of potentiometers to sense the wheels rolling over a desk surface. Soon, mice used a ball and a pair of optical encoders to convert the movement of the hand into digital signals indicating precise relative motion. Now, even the most inexpensive mice use a specialized camera and image-processing algorithms to sense motions at the scale of one one-thousandth of an inch several thousand times per second. Accelerometers, devices that sense acceleration due to motion and the constant acceleration due to gravity, are another interesting example. Today's tiny accelerometers were originally developed for application in automotive air-bag systems. Digital cameras now incorporate accelerometers to sense whether a picture is taken in landscape or portrait mode, and save the digital photo appropriately. Many laptops with built-in hard disks also include accelerometers to detect when the laptop has been dropped, and park the hard drive before impact. Meanwhile, mobile phone manufacturers are experimenting with phones that use accelerometers to sense motion for use in interaction, such as in-the-air dialing, scrolling, and detecting the user's walking pattern.

Research in human–computer interaction (HCI) explores the application of sensing to enhance interaction. The motivation of this work is varied. Some researchers seek to either expand the array of desktop input options or build completely new computing form factors such as mobile devices that know where they are pointed and intelligent environments that are aware of their inhabitants. Other researchers are interested in using sensors to make our machines behave more like we do, or alternatively to make them complement human abilities. Entertainment, surveillance, safety, productivity, mobile computing, and affective computing are all active areas in which researchers are applying sensors in interesting ways.

Although a wide array of sensors is available to researchers, rarely does a sensor address exactly the needs of a given application. Consider building into a computer the capability to sense when its user is frustrated. Detection of user frustration would allow a computer to respond by adopting a new strategy of interaction, playing soothing music, or even calling technical support; however, today, no "frustration meter" may be purchased at the local electronics store. What are the alternatives? A microphone could be used to sense when the user mutters or yells at the machine. A pressure sensor in the mouse and keyboard could detect whether the user is typing harder or squeezing the mouse in frustration (Klein, Moon, and Picard 2002; Reynolds 2001). A webcam might detect scowling or furrowing of the eyebrows. Sensors in the chair could detect user agitation (Tan, Slivovsky, and Pentland 2001). Ultimately, the system chosen should probably exploit a consistent, predictable relationship between the output of one of these sensors and the user's frustration level; for example, if the mouse is squeezed at a level exceeding some set threshold, the computer may conclude that the user is frustrated.

In our effort to build a frustration detector, we may find a number of issues confounding the relationship between the sensors and the state to be detected:

- There is no easy a priori mapping between the output of the sensors and the presumed state of frustration in the user. Implementation of a pressure sensor on the mouse requires observation of the user over time to determine how much pressure reliably indicates frustration. Implementation of the more complex approach of detecting furrowed brows by computer vision requires an elaborate image-processing algorithm.
- The output of the sensors is noisy and often accompanied by a degree of uncertainty.
- Initial experimentation reveals that while no single sensor seems satisfactory, it sometimes may suffice to combine the output of multiple sensors.
- Our preconceived notions of frustration may not correspond to what the sensors observe. This may cause us to revisit our understanding of how people express frustration, which, in turn, may lead us to a different choice of sensors.
- The manner in which the user expresses frustration depends greatly on the user's current task and other contextual factors, such as the time of day and level of arousal. Exploiting knowledge of the user's current application may address many cases where our algorithm for detecting frustration fails.
- After realizing that our frustration detector does not perform flawlessly, we struggle to balance the cost of our system making mistakes with the benefit the system provides.

These are just some of the considerations that are typical in a nontrivial application of sensors to recognition in HCI. Although this chapter does not propose to solve the problem of detecting and responding to user frustration, it will survey aspects of sensor-based recognition highlighted by this example. In particular, this chapter presents the variety of available sensors and how they are often used in interactive systems. Signal processing, recognition techniques, and further considerations in designing sensor and recognition-based interactive systems are briefly addressed.

7.2 SENSORS AND SENSING MODES

This chapter focuses only on those sensors relevant to interactive applications and their typical modes of use. Experimenting with such sensors has never been easier. Microcontrollers such as the Microchip PIC and BasicStamp can interface sensors to PCs and other devices, and can be programmed using high-level languages such as C and BASIC. The Phidgets hardware toolkit (Greenberg and Fitchett 2001) enables effortless "plug and play" prototyping with modular sensors and actuators that are plugged into a base interface board, and provides software application programing interfaces (APIs). The Berkeley and Intel Mote projects also offer wireless sensor packages useful in data collection and sensor networks (Kahn, Katz, and Pister 2000; Nachman et al. 2005). Another source for inexpensive sensors and sensor interface kits is the hobbyist robotics community.

7.2.1 Occupancy and Motion

Probably owing to the importance of sensing technology in security applications, many devices and techniques exist to sense either motion or a person's presence (occupancy). Among these are the following:

- Air pressure sensors that detect changes in air pressure resulting from the opening of doors and windows
- Capacitive sensors that detect capacitance changes induced by the body

- Acoustic sensors
- Photoelectric and laser-based sensors that detect disruption of light
- Optoelectric sensors that detect variations in illumination
- Pressure mat switches and strain gauges
- Contact and noncontact (magnetic) switches
- Vibration detectors
- Infrared motion detectors
- Active microwave and ultrasonic detectors
- Triboelectric detectors that detect the static electric charge of a moving object (Fraden 2003)

Perhaps one of the most familiar motion detectors is the passive infrared detector, which is sensitive to small changes in the pattern of infrared radiation within the spectral range of 4–20 μm (far infrared). Passive infrared detectors sense heat changes over a small duration of time, indicating the presence of a person moving through the room. These devices often control lights in office buildings and can be useful in office-awareness applications when combined with other sensors.

More selective motion and occupancy detection can be obtained with video cameras and simple computer vision techniques. For example, a computer vision system allows for the definition of multiple regions of interest that allow fine distinctions regarding the location of motion. Such a system may thus be able to ignore distracting motion.

7.2.2 Range Sensing

Range sensors calculate the distance to a given object. Such detectors can be used as occupancy detectors, and they are also useful in motion- and gesture-driven interfaces.

Many range and proximity sensors triangulate the position of the nearest object. For example, the Sharp IR Ranger emits a controlled burst of near infrared light from a light-emitting diode (LED). This light is reflected by any object within a few feet and is focused onto a small, linear, charge-coupled devices-array that is displaced slightly from the emitter. The position of the reflection on the sensor can be related with the distance to the object by trigonometry. Similar approaches can be used over a longer effective distance with the use of lasers rather than LEDs.

Stereo computer vision systems similarly use triangulation to calculate depth. If the same object is detected in two displaced views, the difference in their sensed two-dimensional (2D) positions, called “disparity,” can be related to the depth of the object (Forsyth and Ponce 2002; Horn 1986). Stereo vision techniques may be used to determine the depth of a discrete object in the scene, or to compute depth at each point in the image to arrive at a full range image.

A second approach to calculating range is based on measuring the time of flight of an emitted signal. The Polaroid ultrasonic ranging device, for example, was originally developed for autofocus cameras, and subsequently became popular in robotics. Such sensors emit a narrow ultrasonic “chirp” and later detect the chirp’s reflection. The duration in time between the chirp and the detection of the reflection is used to calculate the distance to the object. Ultrasonic range finders can sometimes be confused by multiple reflections of the same chirp; such difficulties are eliminated by measuring the time of flight of emitted light rather than sound, but such sensors are still comparatively exotic.

7.2.3 Position

Designers of sensing-based interactive systems would probably most like a low-power, wireless, inexpensive three-dimensional (3D) position sensor that does not rely on the installation of complicated infrastructure. Originally designed for military application, global positioning satellite (GPS) devices are useful for sensing street-level movement but are limited to outdoor application. Unfortunately, no indoor tracking standard has gained the popularity of GPS.

The position of a wireless RF receiver can be determined by measuring signal strengths to RF transmitters of known position using Wi-Fi, Bluetooth, and Global System for Mobile communications (GSM) standards (LaMarca et al. 2005). Under the assumption that signal strength approximates distance, position may be determined by triangulation, but often interference from buildings, walls, furniture, and even people can be troublesome. Another approach is to treat the pattern of signal strengths as a “signature” which, when recognized later, indicates the position associated with the signature (Krumm and Horvitz 2004). Using Wi-Fi transceivers and a number of Wi-Fi access points, position can be calculated within several feet of accuracy under ideal laboratory conditions using a combination of approaches (Letchner, Fox, and LaMarca 2005). Finally, the Ubisense location system achieves accuracy on the order of 15 cm indoors by using arrival time and angle of RF signals in ultrawideband frequencies.

Commercially available motion capture systems use a variety of strategies to track the position and orientation of multiple points. These systems are generally used to record human motions for applications such as video game character animation. Most require the performer to wear several small tracking devices. For example, when precisely calibrated, the Polhemus and Ascension magnetic tracking devices achieve millimeter accurate, six degrees of freedom (position and orientation) tracking of multiple points, but rely on technology that connects each tracking device to a base station with wires. Such products have been very useful in prototyping gesture-based interactions that require accurate 3D position and orientation (Fitzmaurice, Ishii, and Buxton 1995; Hinckley et al. 1998; Ware 1990; Ware and Jessome 1988), but often are too expensive for widespread use.

Much research in computer vision focuses on accurate and reliable object tracking. Where multiple cameras are available, it is possible to compute the 3D position of a tracked object using triangulation. To recover 3D position useful for interactive applications in a typical room setting, such cameras require careful calibration. Several prototype interactive systems use vision techniques to track the hands, head, and body of a user. Often, a model of shape and appearance that includes typical skin color is used to track the head and hands. The ALIVE system, for example, determines the 2D position of the head and hands of a user by first extracting the user's silhouette against a controlled (static) background. Point of high curvature along this contour are then extracted and tracked as hands or head. Later variants exploit color information as well. Depth is computed by assuming the user is standing on a flat floor (Maes et al. 1995; Wren et al. 1995).

Computer vision-based tracking systems often suffer from poor tracking reliability and sensitivity to variations in background illumination. Tracking reliability can be enhanced by controlling the appearance of the object so that it can be tracked unambiguously. A number of professional motion capture systems, such as the Vicon Peak system, rely on passive, wireless, infrared-reflective pieces, but also require a powerful infrared light source and multiple, redundant sensors (cameras) to minimize missing data resulting from occlusion. Alternatively, cameras sensitive in the infrared domain can be used to track an infrared LED (IR-LED). The position sensitive device, for example, is an inexpensive, camera-like device that reports the brightest spot on its imaging array and is thus suitable for inexpensive IR-LED-based tracking systems. Multiple IR-LEDs can be tracked using a position sensitive device by carefully controlling when each is illuminated. Gross room-level location can be determined using the simplest infrared detectors and IR-LEDs that transmit the identity of the user by blinking specific patterns over time, much like a television remote control (Want et al. 1992).

Acoustic tracking systems are able to triangulate position using time-of-flight measurements. One approach is to equip the room with multiple detectors that are able to hear a mobile tracking device equipped to emit a sound at a known frequency. This configuration can also be inverted, with the detector on the tracked device and the emitters in the environment (Smith et al. 2004; Ward, Jones, and Hopper 1997). Related signal-processing algorithms can combine the output of two or more microphones to triangulate the position of an arbitrary sound source (Rui and Florencio 2003). This approach can be particularly effective when combined with other techniques such as computer vision-based face tracking (Zhang and Hershey 2005).

7.2.4 Movement and Orientation

Unlike most tracking technologies, a number of movement and orientation sensors do not rely on external infrastructure. Inertial sensors, for example, sense spatial and angular motion (translation and rotation). They can be used for activity recognition as well as gesture and body motion-based interactive applications where it is acceptable to wear or hold a small wireless sensor package (Bao and Intille 2004; Hinckley et al. 2000; Lee and Mase 2002).

Very simple tilt sensors such as mercury switches have been used for years to sense gross orientation. More recently, inexpensive accelerometer devices using micro-electromechanical systems (MEMS) technology were developed for application in automotive airbag systems (Kovacs 1998). MEMS accelerometers feature a tiny proof mass or cantilever beam and deflection sensing circuitry to sense both varying accelerations due to movement as well as the constant acceleration due to gravity. Two-axis MEMS accelerometers can be applied to sensing tilt (pitch and roll) and have been used in gaming controllers. But nonmilitary accelerometers are not sufficiently precise to support the double integration of acceleration necessary to calculate position information for more than a few seconds. In such applications, it may suffice to add a coarse position sensor to combat the effects of drift.

MEMS technology has also allowed the development of gyroscope devices that sense angular acceleration rather than absolute orientation. These devices have been used in stabilizing handheld cameras and in the GyroMouse product, which maps relative change in gyroscope orientation to the relative motion of the mouse cursor.

Magnetometers are compact, solid-state devices able to detect the strength of the earth's magnetic field along its principle axis, and so are useful in determining absolute orientation information. The output of a pair of orthogonally mounted magnetometers held level may be combined to find magnetic north. It is common to combine a two-axis accelerator with a two-axis magnetometer to "correct" the output of the magnetometer when it is not held level (Caruso 1997). Three-axis magnetometers are available, but alone do not give a true 3D orientation (e.g., a magnetometer's reading does not change when it is rotated about magnetic north).

7.2.5 Touch

The microswitch typical of today's mouse requires a certain amount of force to activate, thus allowing a user to comfortably rest their forefinger on the button without accidentally clicking. Pressure sensors, however, sense a continuous range of pressure states. Historically, these have been useful in robotics, where they play an important role in designing control systems for manipulators. Polyvinylidene fluoride films and force sensitive resistors are two inexpensive types of pressure sensors with good dynamic range and form factors useful for small devices and interactive systems. Flexible strain gauges utilizing the piezoresistive effect have a resistance related to the amount of deformation (bend) applied to the sensor. Such gauges have been used as the basis for inexpensive glove devices that sense the deflection of each of the fingers of the hand.

Capacitive sensing is based on the property that nearly any object is capable of storing electric charge, and that charge will flow between two objects when touching or in close proximity. A zero-force touch sensor can be implemented with a charge transfer technique, whereby the capacitance of an electrode is estimated by measuring the time taken for an electrode to discharge a small applied charge (Hinckley and Sinclair 1999). This time drops dramatically when the user places a finger on the electrode, since the user's body takes on much of the charge. Other capacitive sensing techniques can sense an object before it touches the electrode, making them suitable as proximity sensors for a wide variety of interactive applications (Baxter 1996; Smith 1999; Vranish, McConnell, and Mahalingam 1991). Multiple electrodes can be used to implement position sensitive sliders, such as the wheel on the Apple iPod.

Most common touch screens report the single 2D position of the user's finger touching or pressing the screen (Sears, Plaisant, and Shneiderman 1992). Resistive touch screens use two large transparent conductive overlays that vary in resistance over their length. When the user presses on the screen, the overlays are brought into contact, and a voltage applied to one or the other overlay is used to detect the horizontal or vertical position of the finger. Capacitive touch screens use capacitive sensing to sense touch, and the relative difference in the charge sensed at each corner of the screen to determine position. Recently, the ability to more precisely sense the touch location and also sense the area and shape of the touching object has been enabled by embedding multiple capacitive sensors in the display surface (Dietz and Leigh 2001; Rekimoto 2002). Finally, surface acoustic wave systems rely on sensing the finger's disruption of surface acoustic waves applied to a screen surface (Pickering 1986).

Finally, computer vision techniques have been applied to sense touch (Fails and Olsen 2002; Han 2005; Matsushita and Rekimoto 1997; Smart Technologies, Inc., 2007; Tomasi, Rafii, and Torunoglu 2003; Wellner 1993; Wilson 2005). Using computer vision to sense touch over an area has a number of advantages: First, these techniques usually do not require a special instrumented surface as do most touch screens. Second, vision techniques naturally support multiple touch points. Finally, vision techniques enable the ability to detect and recognize a variety of objects besides fingers. For example, barcode-like visual codes may be applied to uniquely identify objects such as game pieces placed on a surface.

7.2.6 Gaze and Eyetracking

Gaze detection refers to determining where a person is looking and is principally the domain of computer vision. It is possible to very coarsely determine head orientation using techniques related to face detection (Wu, Toyama, and Huang 2000), but head orientation is often a poor indicator of where someone is looking. The goal of eyetracking systems is to precisely determine where the user is looking, or foveating. Usually, these techniques are based on precise tracking of multiple reflections of an infrared illuminant off the eye's cornea. For good performance, however, eyetracking systems require careful per-user calibration, and so have seen limited general application in interactive systems (Beymer and Flickner 2003; Jacob 1993; Tobii 2005; Zhai, Morimoto, and Ihde 1999).

Rather than determining gaze in a general fashion only to later match the gaze direction to one of several known objects, an alternative is to determine only whether user is looking at the object. The detector can then be embedded in the object itself. Furthermore, the reflection of an infrared illuminant by the cornea and retina can be detected by simple image-processing techniques when the camera and infrared illuminant are colocated (Haro, Flicker, and Essa 2000; Shell et al. 2004).

7.2.7 Speech

The long history of research on speech recognition techniques has resulted in commodity systems that bring modern speech recognition to anyone with a PC and an inexpensive microphone (Rabiner and Juang 1993). New interactive systems, however, highlight the need for further work. Current systems, for example, function poorly without a "close-talk" microphone and in noisy environments, and so are unsuited for use in such contexts as intelligent rooms and mobile scenarios.

The array microphone combines audio from multiple microphones to address the problems of multiple sound sources and noisy environments. Through the process of beamforming, the outputs of the multiple microphones of an array are combined to form a single audio signal in which all but the dominant speaker's signal has been removed. Beamforming can also reveal information about the position of the speaker (Tashev and Malvar 2005).

To achieve robustness, speech may also be combined with other input modalities such as pen gestures (Oviatt 2002). Such approaches usually require a sophisticated model of the user's interaction. Perhaps inspired by HAL in *2001: A Space Odyssey*, some researchers have proposed incorporating computer vision-based lip-reading techniques into the speech interpretation process (Stork 1998). Finally, information such as intonation, prosody, and conversational turn taking can be valuable in interactive systems (Bilmes et al. 2005; Choudhury and Pentland 2003; Pentland 2004).

7.2.8 Gesture

Many notions of gesture exist in interactive systems, and thus, many sensor systems are applicable. A gesture can be thought of as a specific hand pose, a spatial trajectory of the hands or stylus, pointing or other motion to indicate an object, or the quality of a motion of almost any body part as it relates to a given application context (McNeill 1992).

Many of the earlier-mentioned tracking and movement sensing technologies have been applied to sense and recognize gestures. For example, a wireless sensor package with multiple accelerometers or gyros can capture motion information useful in recognizing many gestures. Computer vision techniques also can be used to track body parts such as the hands and head, as well as overall motion qualities that can be interpreted as gesture. Often such systems ease the sensing task by requiring the user to wear brightly colored gloves, or by training precise models of skin color (Brashear et al. 2003).

Pen gestures are often studied on the tablet computer, a form factor that often uses the Wacom electromagnetic positioning technology. This system uses coils embedded in the pen and under the display to find pen position, limited height (hover) above the surface, tilt, pressure, and button state. It can also support multiple simultaneous inputs. The gestures themselves are usually modeled as simple "flick" gestures or spatial trajectories.

Although computer vision techniques have been explored to recover detailed hand pose information, gloves with built-in sensors are more commonly worn for this purpose (Baudel and Beaudouin-Lafon 1993). Early virtual reality systems, for example, used magnetic trackers attached to gloves equipped with bend sensors to recover the position, orientation, and pose of the hands. More recently, vision-based professional motion capture systems that track infrared retro-reflective balls have been used in a similar fashion. With such precise hand shape information, it is possible to point at an object with the index finger, and then make a motion similar to pulling the trigger of a gun to effect an action (Vogel and Balakrishnan 2005).

7.2.9 Identity

In interactive systems, it is often useful to know the identity of an object or user, and a variety of sensing systems are designed to recognize known objects. Object recognition is an active research area in computer vision. There are practical techniques for quickly recognizing one of many known flat objects such as photos or book covers, for example (Lowe 2004). Computer vision-based face recognition techniques have also been shown to work in fairly controlled settings (Li and Jain 2005). However, general object recognition and face recognition in uncontrolled settings is still difficult.

Beyond face recognition, biometrics uses a variety of sensing technologies. Fingerprint recognition hardware, for example, uses optical scanning technology, or an array of tiny capacitive sensors, to construct an overall picture of the fingerprint. Since Johansen's early experiments demonstrating an ability to recognize human motion from point-light displays (Johansson 1973), researchers have worked on gait recognition techniques from video (Boyd and Little 2005). Iris, retina, hand geometry, vascular pattern, handwritten signature, and voice dynamics are other biometric techniques using sensing technology (Sugiura and Koseki 1997; Wayman et al. 2004).

In the absence of reliable techniques to recognize an object by its natural properties, it is often useful to "tag" an object with a standard, easily recognizable marker that reveals the object's identity. Visual codes such as the ubiquitous UPC bar code symbols, for example, are read by laser scanning systems, which are now small enough to be incorporated into mobile devices. Two-dimensional "matrix codes" such as the Quick Response (QR) code pack more bits into the same space, and they have been used in a variety of interactive systems that recognize them by image analysis (Kato et al. 2000; Rekimoto and Ayatsuka 2000).

Recently, radio frequency identification (RFID) tags have gained in popularity. RFID tags themselves are usually passive, can be made small and unobtrusive, are cheap to manufacture, and can be read at a distance. A scanning antenna that emits an RF signal reads the tags; this signal, in turn, powers the tags with enough energy to respond with an identification code. RFID systems vary in terms of range, power requirements, antenna and tag form factors, bit depth, and so on (Garfinkel and Rosenberg 2005). They are thus particularly attractive for commercial inventory management applications and have been applied to interactive systems (Want et al. 1999).

7.2.10 Context

Sensors can provide important information about the context of the user or device. For example, a computer may listen in on an office to determine whether a meeting is in progress, and if so withhold noncritical notifications (Oliver and Horvitz 2005). A later section explores the role of context in interpreting the output of sensors.

Simple context sensors are especially useful in mobile applications. Environmental sensors that detect such information as air temperature, lighting quality, and air pressure may be more directly relevant to the application than the absolute location given by a GPS sensor (Lester, Choudhury, and Borriello 2006; Schmidt, Beigl, and Gellersen 1999). Context sensors may be used to determine the user's activity, or what the user is currently doing. In mobile applications, an inertial sensor may be used to determine the current transportation mode of the user, while a microphone may be used to conclude that the user is engaged in a conversation. An array of simple switches placed throughout a household environment, such as on kitchen cabinets and drawers, may be all that is needed to reliably determine the activities of its inhabitants (Tapia, Intille, and Larson 2004; Wilson and Atkeson 2005).

7.2.11 Affect

In psychology, affect refers to an emotion or subjective feeling. Recently, there has been interest in applying sensing technology to allow interactive systems to respond appropriately to (and perhaps influence) the user's affect (Picard 2000). A system might respond to the user's boredom, interest, pleasure, stress, or frustration (as in the example in the introduction) by changing aspects of the interaction.

Like other multimodal systems, an affective computing system is likely to integrate a variety of conventional sensors. There is an emphasis, however, on the use of physiological sensors to recover physical data that may be related to the user's affective state. For example, the galvanic skin response sensor measures the skin's conductivity, which increases quickly when the user is startled or experiences anxiety. The blood volume pulse sensor measures blood pressure over a local region by measuring the reflectance of a bright infrared light, and can detect certain states of arousal when applied to the fingertips. Respiration rate can be sensed by measuring the amount of stretch in an elastic band worn around the chest. The electromyogram sensor measures the amount of electrical activity produced when the muscle it is placed over contracts, and is useful in detecting jaw clenching, and contraction of various muscles related to facial expressions. Finally, the electrocardiogram measures heart rate.

In designing affective computing systems, it is often difficult to determine the mapping of sensor outputs to application-specific quantities, such as emotional state. Particularly challenging is the task of identifying specific physical correlates for broadly defined emotional states such as "frustration." Finally, physiological sensors are unsuitable for many applications because of the difficulty in deploying them: many must be placed on particular locations on the body, may require good contact with the skin, and are susceptible to differences among individual users or even the same user from day to day.

7.2.12 Brain Interfaces

Advances in cognitive neuroscience and brain imaging technology have spurred initial explorations into interfacing computers directly with a user's brain activity. Much of the work is motivated by a desire to help individuals who have lost the motor skills necessary to use traditional interfaces. Thus, the goal of brain–computer interfaces (BCI) is often to enable users to explicitly manipulate brain activity in order to provide input to a system. Such interfaces typically emulate traditional interfaces by triggering keystrokes and cursor control. However, future applications will likely take advantage of the unique abilities of BCI systems to enable completely new styles of interaction (Hjelm and Browall 2000).

BCI is generally limited to brain imaging techniques that are noninvasive and do not require bulky, expensive equipment. The electroencephalograph (EEG) measures electrical activity at local parts of the brain using electrodes placed carefully on the scalp. EEG has low-spatial resolution compared with other brain imaging techniques, but has relatively good temporal resolution. Functional near infrared imaging measures blood flow in local regions of the brain by calculating the absorption of infrared light directed into the scalp. The technology suffers, however, from low-temporal resolution, but obtains higher spatial resolution than EEG and generates results that are similar to more impractical blood flow–related imaging techniques such as functional magnetic resonance imaging.

With today's BCI systems, users must learn how to manipulate their brain activity effectively for a given application, either through operant conditioning or by executing certain predetermined cognitive tasks that are distinguishable to the sensors. Imagining the performance of a motor skill, for example, exercises specific parts of the brain (Curran and Stokes 2003). This specific activity may be detectable by an imaging technique of coarse resolution. Alternatively, applications can be specifically designed to take advantage of naturally occurring brain activity, such as that associated with a flashing light. Under carefully controlled conditions, it is possible to classify the user's engagement in cognitive tasks, such as rest, mental arithmetic, and mental rotation (Kiern and Aunon 1990).

7.3 SIGNAL PROCESSING

It is rare to find a sensor precisely suited to a given sensing task. Often, sensor output must be manipulated or combined with other sensors to fit the needs of the application. This section surveys signal-processing techniques useful in applying sensors to input and recognition tasks.

7.3.1 Preprocessing

Preprocessing refers to the earliest stage of processing sensor signals. It is at this stage that noise may be removed from raw sensor signals, or signals may be reduced to make them more compact and otherwise easier to use in later processing stages.

The performance of a sensor relevant to preprocessing can be characterized in several ways. Accuracy refers to the degree to which the sensor readings represent the true value of what is measured. Precision, or resolution, by contrast, refers to the extent to which successive readings of the same physical phenomenon agree in value. It is important to realize that while a device's resolution is often measured in bits, this number is often distinct from the number of bits used to represent or store the sensor's readings.

In general, accuracy and precision can be estimated by collecting several successive measurements (samples or observations) of the same input, and computing the resultant mean and scatter (covariance). An accurate sensor will put the mean near the true value, and a precise sensor will have a small amount of scatter about the mean. An accurate but noisy sensor will have low precision (high scatter), but can still be useful by the Central Limit Theorem from statistics: If we make some assumptions about the noise and average a sufficient number of successive values, then we will derive a good estimate of the true value (Hoel, Port, and Stone 1971).

Averaging of successive sensors readings is but one simple way to smooth noisy data to obtain a noise-free estimate of what is measured. Of course, the input in a real application is likely to be changing over time, and the manner in which this average is computed can vary. For example, the boxcar filter is simply the average of the last *n* samples, and is thus easy to implement; however, the boxcar filter suffers because it requires a buffer of samples over which the average is computed, and the resulting estimate will lag the true value in the case of a changing signal. Related to the boxcar filter is a technique whereby the estimate is obtained as a weighted average of new observation and previous estimate. This filter is even easier to implement and requires no buffer of previous samples. In this technique, however, each estimate depends on previous estimates as the Poisson distribution over time, such that a very quickly moving signal or a signal with many outliers will result in erratic changes in the smoothed signal.

The Kalman filter is a popular technique for filtering time-varying signals and can be used to both smooth and predict a signal. It is the optimal linear filter in the sense that it minimizes the difference between the estimate and the true value, assuming a linear model of the input's changing signal and

Gaussian noise (Welch and Bishop 2004). The most common Kalman filter for estimating the position of a moving object models the object's state as a linear function of both velocity and the position estimate in the previous time step. The Kalman filter models uncertainty in two ways. First, there is some uncertainty in the linear model (how much do we believe that the linear model is correct?). Second, there is uncertainty resulting from instantaneous noise that corrupts the observation (how precise is the sensor?). A properly tuned Kalman filter balances these uncertainties appropriately and suffers fewer problems with lag than, for example, a boxcar filter. When the changing signal is actually linear, it can completely defeat lag due to filtering. An improperly tuned Kalman filter, however, can impart unnecessary lag and overshoot, such that the estimate runs past the input before correcting itself.

Often, rather than obtaining a continuous estimate of a sensed quantity, we are interested only in obtaining a binary result: Is the switch on or off? When a user throws on a switch in a real system, the output state of the switch can change very rapidly from off to on and back several times before settling to a single stable state. Debouncing techniques combat this effect; one simple technique is to ignore the switch for some small, fixed time after seeing the first change in switch state (e.g., 40 milliseconds).

More difficult is the situation in which a truly continuous quantity must be transformed into a binary signal. This is commonly done by choosing a threshold below which we report the output as "zero," and otherwise, "one." For example, in using a continuous-valued tilt sensor such as an accelerometer to determine whether a tablet PC is being used in "portrait" or "landscape" mode, it is necessary to transform the tilt information into a binary quantity indicating "portrait" or "landscape." An alternative to a single threshold is a scheme with two thresholds and a region between (a deadband) in which no change to the output is made. Similar to debouncing, this approach can prevent fluctuation of the output around a single threshold.

Choosing threshold values is generally challenging, and poorly designed complex systems are frequently awash in thresholds that require modification to achieve acceptable performance. Ultimately, the thresholding process destroys information; depending on the nature of subsequent processing, this loss can be detrimental to a system's overall performance. This is particularly true for borderline cases where a system is likely to make erroneous decisions as the result of either an improperly chosen threshold or a noisy input. Such concerns may be eased by adopting a "soft" threshold that reports intermediate results around the "hard" threshold; the logistic function (Bishop 1995) can be useful in this approach.

The signal's effective range of output, or dynamic range, must be often be considered both in thresholding and subsequent processing. The relevant range of the property to be sensed must of course lie within the dynamic range of the sensor. If the dynamic range changes—as a consequence, for example, of temperature change, lighting change, or even variations in installation—it may be necessary to calibrate the signal to achieve a normative range. One strategy is to find the sensor's minimum and maximum output during normal use and to map these to some canonical range. For example, the output of a photodetector may be mapped to a range from zero to one by recording the value of the sensor in the darkest and brightest conditions of regular use. Another strategy is to calibrate the sensor to ground truth values that are collected by some other more trusted sensor. Both of these approaches require care if the sensor is not linear in its response; it may then be necessary to fit a curve (e.g., polynomial) to map the sensor to normal values. Characteristics such as dynamic range, linearity of the response, and variation due to temperature are often detailed in a sensor's "data sheet," available from the manufacturer.

In time-varying systems, we are often concerned with the frequency with which we receive new samples from the sensor. An overly high sampling rate can result in too much data to process, and can be reduced by downsampling. By contrast, many interactive systems will seem to lose their responsiveness if the overall latency is greater than 100 milliseconds. Latency or lag refers to any delay present in the sensor's response to a change in the sensed property of the world, and can limit the responsiveness of an interactive system built on the sensor (MacKenzie and Ware 1993). A low-sampling rate imparts latency, which may be remedied by predictive techniques such as the Kalman filter.

Finally, it is important to consider the true distribution of any noise in filtering and many subsequent processing techniques. Many techniques—including the simplest averaging, Kalman filters, and many probabilistic approaches—assume a Gaussian or uniform distribution of noise. Outliers violating this assumption can be troublesome and should be removed by ad hoc means or techniques from the field of robust statistics (Fischler and Bolles 1981; Huber 1981). For example, the median filter, in which values of a sequence are replaced by the median value, is easy to implement, yet more robust than simple averaging.

7.3.2 Feature Selection

In the context of recognition, a feature can refer to a particular sensor or a piece of information derived from one or more sensors, or even derived from other features. Often thought of as a preprocessing step, feature selection refers to the process of determining which features are to be computed from the raw inputs and passed to the next level of processing. Appropriate feature selection can sometimes make difficult recognition problems easy. For example, one somewhat unusual approach to detecting faces in video is to detect eye-blinking patterns. Blinking provides a signal that is easily detected by simple image-processing operations, and is further supported by the fact that both eyes blink together and are arranged in a symmetric spatial configuration on the face. Blinking thus may be highly diagnostic for faces (Crowley and Berard 1997).

Feature selection begins by determining a set of sensors relevant to the task at hand often with knowledge of the task or domain. In the course of development of a new sensing-based system, it can be beneficial to incorporate as many physical sensors as possible, with the idea that subsequent feature selection processes will indicate which sensors are necessary and sufficient. Furthermore, a number of sensors taken in combination may provide the best overall performance.

Having selected a number of sensors, often the next step is to compute derived features from the raw sensor inputs. For example, when an unimportant and unpredictable offset is present in the sensor inputs raw levels, it may be easier to work with its derivative instead. Again, these features are determined in an ad hoc fashion, in light of special domain or application knowledge. For example, early stylus gesture recognition algorithms relied on simple derived features such as the (a) initial angle of the stroke, (b) maximum speed obtained, (c) size of the containing bounding box, (d) duration of the stroke, (e) amount of change in curvature along the gesture, and so on (Rubine 1991). Early face recognition approaches relied on features such as the distances between facial features such as the eyes, nose, and mouth (Zhao et al. 2003). In the domain of audio, the linear predictive coding and the Fourier transform are useful derived feature spaces. The Fourier transform in particular has general applicability to signals with periodicity. For example, the Fourier transform of a body-worn accelerometer may reveal patterns of the user's walking (Hinckley et al. 2000).

Often, it is desirable to put spatial features in a local coordinate system. For example, a gesture recognition system may begin with the position of the head and hands of the user. To remove the effects of the person moving about the room, the head position may be subtracted from the position of each hand, yielding a "head-centric" coordinate system. In the spirit of asymmetric bimanual models of gesture, we might also consider a coordinate system centered on the non-dominant hand (Guiard 1987; Hinckley et al. 1998). In many cases, switching to a local coordinate system eliminates a large source of the irrelevant variation present in the raw signal, thus easing subsequent modeling, and it can be superior to using only derivative information.

If there is a large number of sensors, or if each sensor is of high dimension (e.g., images taken from video cameras), each sensor's value is unlikely to be statistically independent from one another. To remove redundancy and make the input a more manageable size, some form of dimensionality reduction may be used to transform each observation into one of lower dimension. One broad class of techniques involves approximating each sample as the linear combination of a small number of basis functions; the coefficients in this linear combination form the corresponding sample in the new, smaller feature space. Principle components analysis (PCA) is a popular technique and is the optimal linear technique in the mean-square error sense. PCA finds orthogonal vectors ("principle components," or eigenvectors) in the input space as basis vectors, each vector reducing variance (scatter) in the data set. PCA has been used in a wide variety of recognition systems, such as face recognition from images, where often less than 50 components are necessary to perform recognition (Pentland, Moghaddam, and Starner 1994). Today, there are numerous techniques related to PCA, many of which are more suited to classification (Fodor 2002).

Where the number of input features is not large, automatic feature selection techniques may be used to determine the subset of features that matter. Although the topic is an active area of research, one technique of general applicability is cross validation (Bishop 1995; Mitchell 1997). The simplest form of cross validation is the holdout method, which begins by dividing the data set into two halves. Several variations of the model are then trained on one half of the data and tested on the other. The variation with the best performance on the test set is selected as the best model. In the case of feature selection, each variation uses a particular subset of the original input features; after trying all such subsets, we are left with the best performing subset of features. For more than a handful of original features this approach will be impractical, so various greedy approximations are often used, such as starting with the full set and eliminating one at a time, or successively adding features from a small set.

7.3.3 Classification and Modeling

Classification refers to the process of determining which of several known classes a given sample or observation is drawn from, and is typically the means by which a novel input is recognized. A classifier can be used, for example, to recognize which of several known gestures the user has performed by the motion of the pen on a tablet. Detection refers to determining the presence of an observation drawn from a known class against a background of many other observations. The distinction between classification and detection is often rather semantic. For example, a face detection system will determine if there is any face present in an image, while a face recognition system will determine the identity of the detected face. Although both operations can be thought of as classification, often they call for different techniques.

When simple thresholding or feature selection operations are not enough to transform a group of sensor readings into a signal that is readily consumed by the application, it is often necessary to exploit more sophisticated classification and modeling techniques. These techniques are particularly useful in cases where it is necessary to use many sensors together, and when there are dependencies among them that are difficult to untangle by simple inspection. Modeling refers to the choices in representation of sensor values, their dependencies, and the computations performed on them.

There are many ways to classify a new sensor observation as belonging to one of several known classes. Approaches in which a model is trained automatically from a set of training examples are the most relevant to sensor-based systems. These techniques are typically the domain of machine learning. The canonical introductory technique is Fisher's linear discriminant (Bishop 1995) in which a closed-form training procedure determines a line in the feature space that optimally divides two classes of training data. A new, unseen sample may then be classified by determining which side of the line the sample lies. Beyond the two-class case, samples are often classified by computing the likelihood that the sample was drawn from each class, and choosing the class with the largest likelihood. Assuming a new observation x, and classes C_i we choose C* as the maximum value $P(C_i)P(xfC_i)$. The prior $P(C_i)$ indicates our belief that a sample is drawn from a class before we even record it, and is often ignored. There are a variety of techniques to derive such probabilistic models from a set of examples.

Common to all these approaches is the ability to characterize the quality of a recognition result. A sample that is correctly classified as belonging to a given class is a true positive. A sample that is incorrectly classified as belonging to the class is a false positive. A sample that is correctly classified as not belonging to a given class is true negative, while a sample that is incorrectly classified as not belonging is a false negative. In the context of interactive systems, a false negative might correspond to when the user provides an input and the system fails to recognize it. A high false-negative rate can lead to an overall impression of unresponsiveness, or a sense on the part of the user that they are doing something wrong. False positives, however, may correspond to when the system takes an action when the user had no such intention, and can lead to an impression that the system is erratic or overly sensitive (Zhai and Bellotti 2005).

In most situations, a clear trade-off exists between the rate of true positives and false positives. Lower the bar for acceptance to increase the true positive rate, and the rate of false positives is likely to increase. In the context of interactive systems, this tradeoff is especially important to consider when developing criteria for when the system takes action as the result of a recognition process. The receiver operator characteristic (ROC) curve plots true-positive rate against false-positive rate and best characterizes this trade-off (see Figure 7.1). The ROC curve is also an established method to compare the performance of recognition techniques, without regard to any application-specific choice on how tolerant we are to false positives.

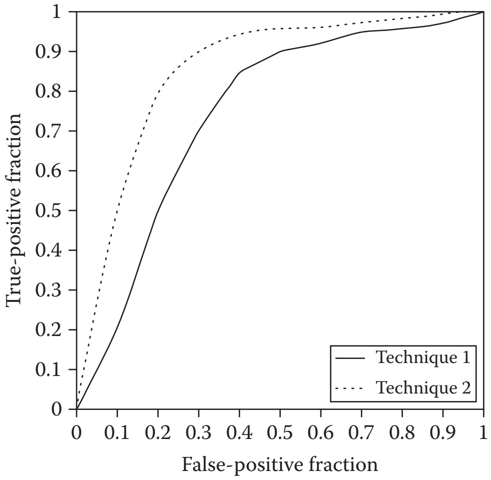


FIGURE 7.1 Receiver operator characteristic curves illustrate the trade-off between the rate of true positives and false positives, and can be useful in comparing recognition techniques. Here we see that for a given tolerable rate of false positives, Technique 2 yields better recognition performance than Technique 1.

In a given application, it is also instructive to break out classification performance by each class. The confusion matrix summarizes how a labeled test set is classified by each class, and may reveal that much of the overall classification error can be traced to errors classifying observations from a small number of classes. This can thus inform design of the classifier or the set of application-relevant categories. Boosting is one technique in which misclassified samples are emphasized in subsequent training of the model to reduce the overall error rate (Schapire 2003).

The naïve Bayes classifier assumes that the value of a given feature is independent of all the others. This property of conditional independence may not actually apply to the data set, but its assumption simplifies computation and often may not matter in practice (hence the label "naïve"). Assuming observations of the form x 5 ^x_1, x_2, …, x_n&, the posterior probability of a class C is $P(Cfx)$ 5 $P(C)\ P(xfC)$ by the Bayes rule. Naïve Bayes treats each feature as independent: $P(Cfx)$ 5 $P(C)P_iP(x_ifC)$. Because each feature is modeled independently, naïve Bayes is particularly suited to high-dimensional feature spaces and large data sets. Each feature can be continuous or discrete. Discrete variables are often modeled as a histogram (or probability mass function), while continuous variables can be quantized or binned to discrete values, or modeled as a Gaussian or other parametric distribution.

A number of other popular classification techniques do not have obvious probabilistic interpretations. The neural network, for example, is best thought of as a function approximation technique. Often, as applied to classification, the input of the approximated function is the observation itself, and the output is a vector whose *i*th component indicates belief that the observation belongs to the *i*th class.

Decision trees can be a powerful classification technique that leads to very compact representations for some problems. Each node of a decision tree corresponds to an assertion about the value of a feature in the observation, and yields a split in the data set. The leaves of the tree then indicate the class to which the observation belongs. Classification of a new sample is then rather like the children's game of "twenty questions." Training the model involves determining how to make the splits to optimize classification performance, and possibly, the size of the tree (Breiman et al. 1984).

The support vector machine (SVM) is a powerful modern alternative to the Fisher linear discriminant (Cristianini and Shawe-Taylor 2000). SVMs determine the split between the two classes to maximize performance on unseen data. Furthermore, SVMs gain much of their power by allowing nonlinear splits of the feature space, but are often thought of as being computational intensive (though, see [Platt 1999]).

Where the conditional independence assumption of naïve Bayes is too strong, other techniques that directly model the joint probabilities are applicable. For example, a mixture of Gaussians uses a sum of multiple Gaussian distributions to model arbitrarily complex joint distributions: $P(xfC)$

5 $S_iP(v_i)$ $P(xfv_i)$, where $P(xfv_i)$ is Gaussian with mean m_i and covariance S_i. Such mixture models may be trained by the expectation maximization algorithm (Mitchell 1997; Neal and Hinton 1999). The expectation maximization algorithm is very similar to clustering approaches such as k-means, in which k points in the feature space are chosen as representative of the overall set of samples.

Often, there are advantages in treating some subset of the variables as conditionally independent from others. For example, a full joint probability distribution can require a lot of data to train; there may be clear constraints from the application that imply conditional independence, and there may be some subset of the variables that are most effectively modeled with one technique while the rest are best modeled with another. In this case, it may be helpful to selectively apply conditional independence to break the problem into smaller pieces. For example, we might take $P(xfC)$ 5 $P(x_1,x_2fC)$ $P(x_3fC)$ for a 3D feature space, model $P(x_1,x_2fC)$ with a mixture of Gaussians, and $P(x_3fC)$ as a histogram. This overall model amounts to an assertion of condition independence between x_3 and the joint space of x_1 and x_2.

This modularity afforded by assumptions of conditional independence is taken to its logical conclusion in the Bayesian network, which is commonly represented as a directed acyclic graph, where each node corresponds to a random variable x_1 with probability distribution $P(x_if\text{parents}(x_i))$, and each variable is conditionally independent of all variables except its parents (Jensen 2001). Nodes in a Bayesian network for which we have observations are called evidence nodes, whereas others are considered hidden. Observations may be entered into network, and through an inference procedure, the likelihood of the observation may be calculated, as well as posterior distributions over any hidden nodes. With Bayesian networks, designers may craft complex probability models without becoming mired in mathematical notation, and software packages allow graphical manipulation of the networks directly (Kadie, Hovel, and Horvitz 2001).

The dynamic Bayesian network (DBN) models time-varying sequences, and thus is relevant to systems in which interactions take place over durations of time. A DBN can be thought of as a Bayesian network where certain nodes depend on the same Bayesian network instantiated on the previous time slice. Dependencies can be within a time slice, or across the time slice. For example, a state variable may depend on its past value as $P(x_tfx_{t21})$. Such relationships with past values of the same variable can encode a probabilistic finite state machine or Markov model, where the distribution $P(x_tfx_{t21})$ is considered a transition matrix. With this dependency on random variables in the past DBNs can effectively encode a time-varying state or "memory" that may be relevant for an interactive application. For example, it can be useful in modeling interactions composed of a sequence of steps, or where application state itself can be modeled as a finite state machine. Finally, by making a strong dependency on the immediate past, the model can be given some inertia or "smoothed."

One popular special case of the DBN is the Hidden Markov Model (HMM), often used to model time-varying signals such as speech, gesture, pen strokes, and so on. HMMs model observations y_t conditioned on a state variable x_t, which evolves over time as $P(x_tfx_{t21})$. As with many probabilistic models, HMMs are generative in nature, meaning one of the ways we can understand them is to consider "running them forward" to generate new observations: an HMM can be thought of as a stochastic finite state machine (Markov model) that generates observations drawn from a distribution associated with each state. With each time step, the Markov model takes a transition to a new state. In the inference (recognition) process, the posterior distribution over the hidden state variable x_t is computed from the observation sequence y_t (Rabiner 1989). HMMs have been applied many types of observation sequences, including hand gestures, handwriting recognition, speech recognition, and so on. In the simplest application paradigm, a separate HMM is trained for each class.

Much of the attraction of Bayesian networks is because of their flexibility to implement complex probabilistic dependencies. Many probability models may be thought of as particular Bayesian networks. Naïve Bayes, mixture of Gaussians, HMMs, and Kalman filters, are among the models that have been shown to be special cases of the Bayesian network (Jordan 1999). The structure of the Bayesian network is often determined by hand using application-specific knowledge, while the various distributions may be tuned by training from data. Parts of a Bayesian network may be completely handcrafted, derived from domain knowledge in the same manner as expert systems. Other sections of the network may in fact be HMMs, Kalman filters, mixtures of Gaussians and various hybrids, such as a mixture of Kalman filters. The automatic learning of the structure of the network itself is active area of research (Heckerman, Geiger, and Chickering 1994).

Many of the techniques outlined in Section 7.3.3 can be applied to the more generic task of modeling, where we are interested in more than classification results. For example, a Bayesian network that fully models the user's interactions with a mobile device might include a variable representing the user's location. The value of this variable will be hidden (unknown) if there is no sensor to directly observe the user's location. We may, however, compute the posterior distribution of the variable after several other kinds of observations are entered in the network and find that the user's location is sometimes known with some precision (e.g., the device recognizes the nearest wireless access point with a known location). Not only is this model useful in deducing the user's location, but also enables other parts of the model to exploit this knowledge even if we are not ultimately interested in location information.

Finally, sometimes the signal-processing task for an application is better thought of as approximating a function that directly maps a set of inputs to outputs. Techniques such as neural networks, radial basis function networks, and other manifold learning techniques can be useful in learning mappings from sensor inputs to application-specific quantities.

Such techniques can be particularly useful in transforming raw, high-dimensional, nonlinear sensor readings into simple calibrated outputs useful in an application. For example, in carefully controlled circumstances, it is possible to map images of a face to gaze angle by providing a number of face image and gaze-angle pairs. A function approximation approach can interpolate over these examples to map new images to gaze angle (Beymer and Poggio 1996).

7.4 EXAMPLE SYSTEM

The following example demonstrates a number of the techniques described in Section 7.3.3 in a working, interactive, sensor-based system. After motivating the overall design of the system, a number of aspects of hardware design are illustrated. Also described are subsequent signal-processing steps such as sensor fusion, the application of Bayesian networks for modeling, and gesture and speech recognition.

In the design of intelligent rooms, the issue of how the room's inhabitants might best interact with the room often arises. The traditional notions of desktop computing or the multiple, incompatible, button-laden remote controls typical of consumer electronics are perhaps antithetical to the seamless and untethered experience that is a main feature of the vision of intelligent environments. One popular notion of how users could control an intelligent room is borrowed directly from *Star Trek*: The user of the room merely speaks to it, as in, "Computer, turn on the lights."

In the development of one intelligent room (Brumitt et al. 2000), a user study was conducted to determine how real users might want to control multiple lights throughout the space (Brumitt and Cadiz 2001). A *Wizard of Oz* paradigm was adopted so that the study would not be limited to designs already implemented. The experimenter, seated behind one-way mirrored glass, operated the lighting controls manually in response to user actions. The users were then exposed to multiple ways of controlling the lights: (a) a traditional graphical user interface (GUI) list box, (b) a graphical touch screen display depicting a plan view of the room with lights, (c) two speech only-based systems, and (d) a speech and gesture-based system. The study concluded that, like Captain Kirk, users preferred to use speech to control the lights, but that the vocabulary used to indicate which light to control was highly unpredictable. This variance in speech chosen poses a problem for the pure speech-based interface.

Interestingly, the majority of subjects looked at the light they were trying to control while speaking. This observation suggests that an intelligent room could resolve ambiguity in spoken commands (e.g., which light to control) by using computer vision techniques to determine the user's gaze, at least where the device under control is within sight. There are a number of general approaches to computing gaze, but each has serious drawbacks. For example, it is possible to roughly compute gaze from a small number of cameras throughout the room (Wu, Toyama, and Huang 2000), but such systems presently lack accuracy and reliability, or require a large number of cameras to cover a useful space. Wearing a special device such as glasses solves some problems, but may not be acceptable to casual users. Another approach is to embed a camera in the device to determine whether the user is looking at it, rather than computing general gaze (Shell et al. 2004). This technique can be effective, but presently scales poorly to a large number of devices.

In light of the difficulties of determining gaze reliably, we reasoned that pointing gestures may play a similar role as gaze in indicating objects. While few subjects in the lighting study spontaneously used gestures, this may be partially explained by the near perfect performance of the *Wizard of Oz* speech recognizer (the experimenter). Furthermore, pointing may have certain advantages over gaze. For example, pointing is typically the result of a conscious decision to take action, whereas changes in eye gaze direction may be more involuntary (Zhai, Morimoto, and Ihde 1999). However, pointing may be no easier to detect by computer vision techniques than gaze (Jojic et al. 2000).

To demonstrate the utility of the combination of pointing and speech as an interface modality in an intelligent environment, we built a hardware device to sense pointing gestures and developed associated signal-processing algorithms to combine speech and gesture (Wilson and Shafer 2003). At the center of the XWand system is a handheld device that may be used to select objects in the room by pointing, and a speech recognition system for a simple command and control grammar. To turn on a light in the room, the user may point the wand at a light and say, "Turn on." Because the pointing gesture serves to limit the context of the interaction (the light), the speech recognition task is reduced to recognizing the few operations available on lights: "Turn on" or "Turn off." Alternatively, the user may perform a simple gesture in place of speech to effect the same command. The user may, for example, point at a media player device, hold the button down, and roll the device to adjust the volume. The XWand system illustrates a number of points related to sensor and recognition-based input, including the hardware design of a composite inertial sensor, sensor fusion, DBNs, and a host of design considerations.

The original XWand system is based on a 3D model of a room and the controllable devices within it. Using onboard sensors, the XWand device can determine its own absolute orientation, while a computer vision system mounted in the environment finds the position of the wand. Given the size and 3D position of an object in the room, it is a simple trigonometric calculation to determine whether the XWand is currently pointing at the object.

The original XWand hardware device contains an Analog Devices ADXL202 two-axis MEMS accelerometer, a Honeywell HMC1023 three-axis magnetometer, a Murata ENC-03 one-axis piezoelectric gyroscope, a 418-MHz FM transceiver, a PIC 16F873 microcontroller, an IR-LED, and a pushbutton mounted on a custom printed circuit board (see Figure 7.2). Although the accelerometer is useful in detecting pitch and

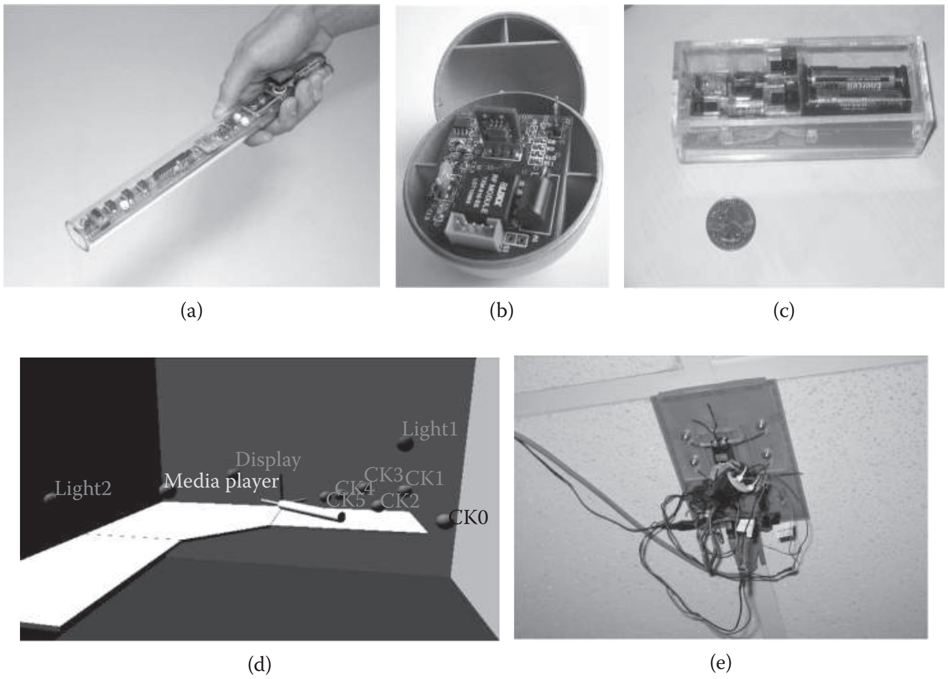


FIGURE 7.2 (a) The first XWand prototype includes accelerometers, magnetometers, gyro, radio, etc. (Image © 2003 ACM.) (b) The Orb device repackaged much of the XWand. (c) The WarpPointer updates most of the components and uses Bluetooth. (d) The XWand three-dimensional geometry model includes the three-dimensional position of all interactive devices in a room. (Image © 2003 ACM.) (e) The WorldCursor teleoperated laser pointer is driven by the XWand.

roll (recall that gravity is an acceleration), it cannot detect the yaw attitude of the device. The three-axis magnetometer reports direction cosines against magnetic north, from which yaw can be determined only if the device is held flat (some GPS devices are equipped with two-axis magnetometers that give heading when the device is held flat). Fortunately, pitch and roll information from the accelerometers may be used to "correct" the output of the three-axis magnetometer to yield a full 3D orientation with respect to magnetic north.

To compute the 3D position of wand, the XWand system uses a pair of FireWire cameras mounted in the corners of the room, which are used to track the IR-LED on the device. Each camera uses an IR-pass filter so that in a typical office environment, only the IR-LED is visible in the image. Furthermore, the IR-LED is programmed to flash at 15 Hz. When the host takes the video output of each camera at 30 Hz, consecutive images may be subtracted pixelwise so that only objects blinking at 15 Hz remain. The IR-LED can thus be located easily in both views. Furthermore, the cameras are calibrated to the geometry of the room so that the 3D position of the IR-LED is obtained from its 2D position in both views. Note that this arrangement assumes a line of sight to the IR-LED from both cameras.

To support speech recognition, an open microphone (low impedance) is placed in the environment. Ultimately, this microphone should be placed on the device, perhaps with the audio encoded and relayed off-board for recognition. The speech recognition engine is programmed with simple command and control grammar based on a simple command-referent pattern, where a referent can be a device in the environment (e.g., a light) and the command refers to one of a number of permitted actions on the device (e.g., "turn on").

Simple gestures made with the wand—such as flicking left, right, up, down, and roll—are recognized by simple routines that measure the change in attitude of the wand from the attitude recorded when the button on the device is first pressed. We have also experimented with using HMMs to recognize more complex gestures, but have chosen instead to exploit a small set of simple, memorable, and reusable gestures in conjunction with other contextual information such as pointing and possibly speech information. This approach allows for a very simple and robust gesture recognition process, and avoids training users on a gesture set of greater complexity.

A DBN fuses the various quantities to arrive at a multimodal interpretation of the user's interaction. It models the combination of the output of the speech recognition, the object at which the wand is currently pointing, any gesture performed, the known state of the devices under control, and the state of the interpretation in the previous time steps (see Figure 7.3). The network bases this combination on the command-referent pattern outlined in the previous paragraph, where the referent may be determined by speech or pointing gesture, and the command may be determined by speech, gesture, button click, or any combination thereof. The ultimate action to be taken (e.g., "turn on light #2") depends on the command and referent, as well as the state of the device itself (e.g., "turn on light #2" is only permitted if the light is off). Finally, both the command and referent at the current time step depend heavily on the command and

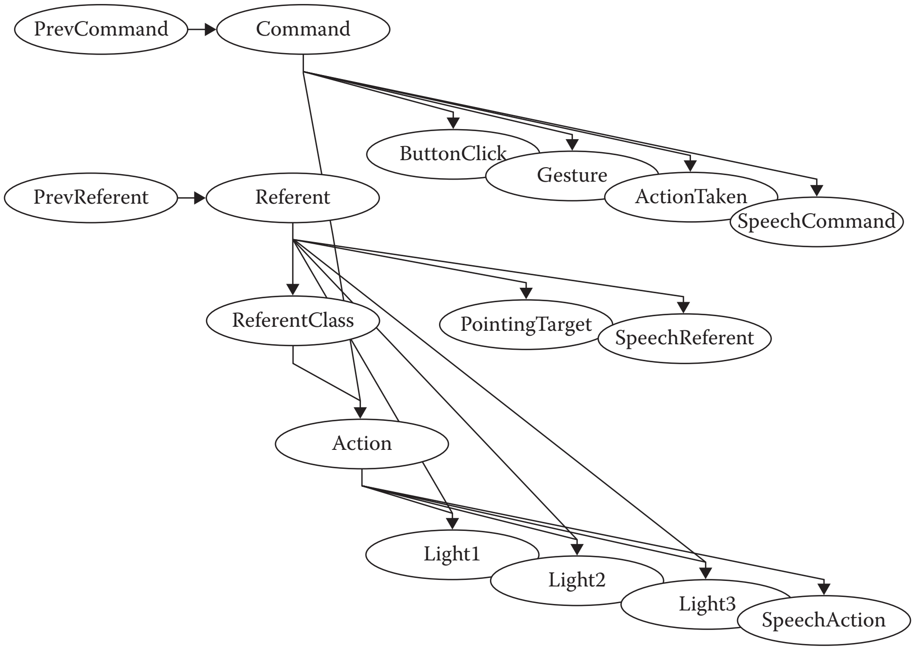


FIGURE 7.3 The XWand Dynamic Bayesian Network models multimodal interaction. It combines wand input (PointingTarget, Gesture, ButtonClick), speech input (SpeechReferent, SpeechCommand, SpeechAction), and world state (Light1, Light2, Light3) to determine the next action (Action) as a combination of command (Command) and referent (Referent) and past beliefs (PrevCommand, PrevReferent). (Image © 2003 ACM.)

referent from the previous time step, such that either quantity can be specified at slightly different moments in time.

The multimodal fusion Bayes network offers a number of interesting capabilities for the XWand system. For example, when the state of each controlled device is represented in the network itself, distributions over related quantities change in appropriate ways. For example, if the user points at a device that is currently turned off, speech and gesture recognition results inconsistent with that state are ignored, such that the phrase "turn off" is removed from the speech recognition grammar. In future work, it may also be possible to infer vocabulary by training the network dynamically (e.g., point at a light and label it as a "light," or point several times at a point in space to train the position of a known light while saying, "Light."

We have continued to develop the XWand prototype in various ways. A typical objection to the camera-based tracking system is the lengthy setup and calibration procedures such systems require. We therefore explored an alternative configuration that eliminates cameras in favor of a single teleoperated laser pointer mounted in the ceiling (Wilson and Pham 2003). The WorldCursor laser (see Figure 7.2) is programmed to match the motion of the wand, in a manner similar to how the standard mouse controls a cursor with relative motion. Because the set of objects the laser can reach is limited by line of sight, the original 3D model of the system is eliminated in favor of a simpler spherical coordinate system with an arbitrary origin, thus simplifying setup.

The most recent iteration includes a three-axis accelerometer that can be combined with a magnetometer to arrive at a true 3D orientation by a simple cross-product calculation (see Figure 7.2). The device has also been applied to cursor control with very large displays, where the mixture of sensors enables a variety of cursor control mechanisms including absolute, orientation-based pointing (position and orientation), relative angular motion similar to the Gyromouse, and pure absolute position only. This flexibility allows exploration of several new modes of interaction. The relative, gyro-based mode of pointing allows very fine control with clutching over a small area. With very large (wall-sized) displays, however, it is easy to lose the cursor among the many onscreen objects. With the current device, it is possible to momentarily adopt one of the more absolute pointing modes to "warp" the cursor to a point directly in front of the user.

7.5 CONSIDERATIONS IN DESIGNING RECOGNITION-BASED SYSTEMS

Section 7.4 touches on issues common to complex sensing-based interactive systems, from hardware design to modeling and sensor fusion. Overall, the strategy in the example is to avoid complex recognition problems whenever possible through thoughtful choice of hardware sensors, design of the interaction, and strong modeling of context. In Section 7.5, we expand upon the motivations behind these design choices. Many of these considerations must be considered in the design of recognition-based interactive systems. While many involve difficult problems with no easy answer, it is best to be aware of these issues when designing a sensing-based interactive system.

7.5.1 Computational Cost, Utility, and the Cost of Failure

A number of challenges exist in designing an interactive system that uses sensors and recognition techniques. First, although Moore's Law continually pushes forward the frontier of practical signal-processing techniques, many of these algorithms are still computationally intensive. Only in the last six years or so have real-time computer vision techniques become practical on commodity hardware. Furthermore, many of the machine learning and pattern recognition techniques are data-driven, thus requiring large amounts of storage, memory, and training. In the case of mobile devices, where computational power lags desktop computing power by several years, many of these algorithms are impractical, and developers are often forced to make hard choices and take shortcuts. These concerns are often magnified in interactive systems that entirely remake the user interface and use multiple recognition-based techniques.

Second, recognition-based systems often face serious challenges achieving and guaranteeing the level of robustness required in real applications. In the case of developing consumer products, for example, it is one thing to demonstrate a technique in a laboratory setting; it is quite another to show that the same technique will work in the variety of circumstances in which customers will expect it to work. Unit testing even simple recognition-based interactive systems can be daunting. Computer vision techniques, for example, are often susceptible to variations in lighting, while audio-based techniques may fail in the presence of background noise. Some effort has been devoted to developing signal-processing techniques that adapt to both the current circumstances and user. Adaptive speech recognition and handwriting recognition techniques, for example, have become commonplace. Even in these cases, however, it is important that the systems have good functionality out of the box or else the user may not use the system long enough for an adaptive algorithm to improve performance.

In the development of a recognition-based interactive system, it may become impractical to seek more improvement in recognition performance. At this point, it is important to consider the cost of recognition failure: Often the cost of repairing a false-positive recognition can overwhelm any advantage in the use of the system. In speech recognition, for example, the repair of the errors can be awkward, slow, and disruptive to the task (Karat et al. 1999). Only users that are unable to use a regular keyboard may accept a dictation system that fails three times out of 100 words, for example (Feng, Karat, and Sears 2005). Another consideration is whether the system returns a result similar to the desired result when it fails (graceful degradation), in which case repair is likely to be easier (Horvitz 1999). If a recognition failure is too costly to consider repair (for example, control of an air lock on a spacecraft, or more mundanely, closing a window on a desktop GUI), the cost of making a mistake may be incorporated directly in the model so that false positives are more likely to be avoided. This can be done either by seeking some kind of deliberate confirmation from the user, or more simply by moving thresholds up the ROC curve. In the latter approach, it is important to be aware that often users modify their behavior on successive attempts such that their input is no longer modeled by the usual training corpus; in speech, this is known as the Lombard effect (Junqua 1993), but this phenomenon can be observed in other modalities as well.

Considerations of computational cost, robustness, and cost of errors should play prominent roles in the end-to-end design of recognition-based systems. Ultimately, the designer may be forced to recast the interaction and possibly the application altogether. An example of a system in which one can find many of these issues played out is the Sony EyeToy, an add-on camera for the Sony Playstation 2. The EyeToy allows the player to interact with a game through gesture and body motion, rather than through the usual game controller. It also appears to use very simple image-processing techniques to determine the player's motion. These techniques are computationally cheap and generally quite robust to varying light levels and other factors that are likely to be encountered in a residential application. Furthermore, the EyeToy works with a small number of games written specifically for use with this device. These games take advantage of the strengths of the EyeToy, rather than risk providing a poor emulation of the regular game controller.

7.5.2 Role of Feedback

Feedback is important in any interactive application, but may be even more so where sensing and recognition are used (Bellotti et al. 2002). We can characterize some kinds of feedback as "tight" or not, where a "tight" feedback loop provides feedback in a frequent, timely, and informative fashion.

Usually, if there is any chance of recognition failure, the system should provide feedback on all recognition results, errors, or otherwise. If possible, the system should provide some indication of the nature of any failure so that the user can modify his or her own behavior to meet the system's expectations. If the system can provide some indication of the quality of the interpretation as it is happening, then it may be possible to allow the user to cancel or modify the interaction on the fly, so as to avoid costly errors or cumbersome confirmation routines (Vogel and Balakrishnan 2004).

The right feedback can often influence the design of the sensing and recognition algorithm itself. For example, because the onscreen cursor is updated so quickly (a tight feedback loop), a naïve user might not ever realize or care that the mouse provides rapid, small, successive bits of relative movement information rather than true position information, which would be much harder to sense. This trick is used in the WorldCursor system to avoid the use of cameras required by the original XWand system.

Considering the EyeToy again, it is interesting to note that the image of the player is often incorporated into the onscreen presentation. By watching themselves onscreen, players are able to interact with onscreen elements without relying on sophisticated, yet more failure prone and computationally intense, hand-tracking algorithms. This feedback also cleverly ensures that the player stays in the camera's field of view; the flow of the game does not have to be broken to alert a player who has left the field of view.

7.5.3 Implicit and Explicit Interaction

In the previous discussion regarding feedback and cost of failure, we assume that interaction is structured in such a way that the user takes action and expects a timely response from the system—that is, the user's actions and the system's responses are explicit. Most interactive systems can be characterized in this way.

In contrast to explicit interactions, implicit interactions are based not on explicit action by the user, but more commonly on users' existing patterns of behavior (Schmidt 2000). For example, with the frustration-sensing system outlined in the introduction, the state of frustration is not explicitly entered by the user in order to elicit some behavior from the system. Instead, the state arises naturally and perhaps involuntarily, and upon detection, the system should take appropriate action.

Implicit interactions may take place over a long duration, and may not exhibit an obvious pattern of cause and effect. For example, systems that adapt to perceived user preferences, as indicated by the history of user behavior, might eventually make a recommendation to the user, or even take actions so subtle that the user may not notice them. Such systems can be complex in terms of sensing and modeling, and often tend toward automating or refining aspects of the user's original task (Horvitz et al. 1998). For example, a smart home may observe its inhabitants' daily patterns of coming and going to determine an optimal schedule to control the thermostat automatically, balancing comfort and economy (Mozer 2005). One potential difficulty is that, unless the output of the system is designed very carefully, users may feel unable to correct a mistake made by the system or exert more explicit control in the face of exceptional circumstances (e.g., a party). Designers should consider providing functionality that allows the user to query the system for why it took a given action, provide simple mechanisms to redress errors, and finally, revert to a manual control mode.

Implicit interaction systems driven by patterns of ongoing user behavior are often not as critically dependent on sensing and recognition reliability and the nature of feedback. Rather, it is more important that interpretation processes are more correct than incorrect over time; accordingly, modeling techniques that integrate noisy sensor values over time are often appropriate. The automatic thermostat, for example, should probably incorporate many observations of the users' behavior—including presence and manual thermostat control—and model weekends, weekdays, and holidays differently. Sophisticated modeling techniques in hand, designers of such systems have the opportunity to exploit powerful, though sometimes unreliable, sensing techniques.

7.5.4 Importance of Context

Notions of context can play an important role in sensing-based interaction (Dey et al. 2001). "Context" refers to the overall environment the interaction or device finds itself in, rather than the objects obviously and directly relevant to the task at hand. What constitutes context often depends on point of view; after studying the application in detail, factors that once may have seemed external and tangentially relevant (context) may be central to the model after all.

Note the purposeful vagueness in the definition of "context"; with respect to sensing, "environment" can refer to the actual physical surroundings. The addition of sensors to a system may give it awareness of its environment, and thereby enable interesting "context-dependent" behavior. For example, a mobile device equipped with GPS or other localization technology might bring up web search results corresponding to the nearest points of interest (Abowd et al. 1997; Davies et al. 2001; Hariharan, Krumm, and Horvitz 2005). Beyond the physical environment, context can refer to more abstract states, such as the user's current activity (e.g., working or not), the history of their interactions, and their preferences. It may also refer to momentary information, such as the action of the nondominant hand, or anaphoric references that provide scope for an ongoing interaction.

Often our activities follow preestablished patterns, either by design or by accident. These patterns can provide strong contextual information for interpreting the user's interactions. For example, entertainment-based scenarios may have a narrative structure that constrains the interaction (Galyean 1995). The KidsRoom interactive experience, for example, was structured according to a narrative progression involving the fantastical transformation of a children's room (Bobick et al. 1999; Bobick et al. 2000) (see Figure 7.4). This structure in turn guided the selection of recognition at any given moment.

Modeling context explicitly can be a powerful way to solve difficult recognition problems. The more context can be brought to bear, often the easier and more robust the recognition. Although counterintuitive at first, by exploiting context, the combination of multiple sensors with simple signal-processing techniques may result in better performance than the use of fewer sensors with more complex signal-processing techniques. Recall how in the XWand system, the speech recognition process is constrained by the knowledge of what the user is currently pointing at with the wand, the current state of the device indicated, and so forth. In fact, the interaction can be so constrained that, often, the system only needs some indication that the user said anything. Similarly, it suffices to use simple, learnable, and reusable gestures when the interaction has been contextualized by pointing, speech, or both. Our frustration detection system may be more robust if it incorporates the knowledge of the application currently in focus. For example, it may be easier to write one frustration detector for office applications and a separate one for use while playing video games, rather than one detector that works in both contexts. In the end, the two detectors may only differ in terms of how some threshold is set.

FIGURE 7.4 The KidsRoom engaged children to participate in an interactive journey. Computer vision and projection technologies were used to transform an ordinary room into a variety of settings including a forest and river. (Courtesy of the Massachusetts Institute of Technology, © 1999.)

7.5.5 Importance of *A Priori* Knowledge

The development of a model for a sensing, interactive system can benefit from specific knowledge of the domain in which the interaction is situated. Such a priori knowledge can lead to insights as to how to determine meaningful categories from raw sensor data. Higher level rules taken from the domain can then be brought to bear. For example, in the case of a pen-based system that automatically parses and manipulates mathematical equations, the rules of how mathematical expressions are combined can be a powerful constraint that drives the correct interpretation of the sloppiest drawings (LaViola and Zeleznik 2004). Similarly, knowledge of chemistry can guide the transformation of sketches of molecules to a full 3D model (Tenneson and Becker 2005). Providing a kind of context, strong assumptions taken from the domain can limit the applicability of a model but can often dramatically improve performance.

Many domains of human behavior have been categorized and described in terms of detailed taxonomies and ontologies. For example, music, gesture, dance, and spoken language each have detailed ontologies, notation schemes, and so on. It can be beneficial to draw from such knowledge when developing a model, but some aspects of a categorization scheme may not be fully supported by the available sensors. For example, there may be some aspect of the domain not covered by the model, or a single category may "alias" to several distinct classes as perceived by the sensors. A detailed analysis of the sensing system output may lead to insights on the original domain model.

One of the advantages of incorporating bits of domain knowledge representation directly into the model itself is that it becomes more transparent to its designers and users, and thus, more reusable. If there is a problem with the system, the designer may directly inspect semantically relevant quantities from the model. Approaches that do not rely on such informed representations, such as neural networks, are often so difficult to inspect that upon discovery of a problem, it may be easier to retrain the model from scratch than to troubleshoot what the network really learned. A good compromise is to use the more data-driven techniques (such as neural networks or probabilistic modeling techniques) to map the raw input signals onto semantically meaningful mid-level primitives. Such choices in representation can support modularity and explorative research.

It is interesting to note, however, that initial research on a given complex sensing problem often draws heavily from domain knowledge, only to be eclipsed later by more purely data-driven approaches. Early approaches to speech recognition, for example, transformed the audio signal into a string of symbols representing phoneme categories developed by linguists; today, one is more likely to see approaches in which subword acoustic unit categories are trained from audio signals directly in a purely data-driven approach. Early face detection and recognition approaches similarly relied on prescriptive or ad hoc features, whereas more recent approaches are more purely data-driven (Li and Jain 2005; Zhao et al. 2003).

7.5.6 Generalize or Specialize?

The incorporation of a priori and context information leads to potentially complex, yet powerful, modeling techniques. One of the drawbacks of adding more and more detail into a model, however, is that the resulting system may be so tailored to a particular domain or set of contextual circumstances that it fails to generalize to new applications. It seems as if engineering best practices, including modularity and device independence, run counter to models that are optimized for a given situation.

For example, consider the problem of delivering various location-based services to inhabitants of an indoor space. A good, familiar choice for the representation of each person's location might be 2D Cartesian coordinates on a full map of the environment. A person-tracking system is installed in the environment; it is charged with determining where everyone is on the map. This information is then passed onto another part of the system that delivers services to each user based on the knowledge of where everyone is at any given moment.

Such a choice of representation has many desirable features. We can consider a multitude of person-tracking technologies—as long as each reports 2D Cartesian coordinates we can incorporate its output into our map (device independence), and we can consider merging the results of multiple such systems in a simple probabilistic framework where we model the location of each person as a distribution over space rather than simple 2D coordinates. Furthermore, we can exploit familiar rules of geometry to calculate interesting properties such as whether a display is within sight of a particular user. Most importantly, we need not concern ourselves with the particular application services that our ultimate system will provide: As long as each service expects only our generic representation, we can expect the service to work properly. The system is general in its applicability.

Contrast this approach to one in which we have no unifying geometric model or map, but instead, install proximity sensors at each of the devices of interest, each sensor detecting someone as they approach the device. Perhaps we install special sensors on some of the doors to detect when someone walks through (Abowd, Battestini, and O'Connell 2003), and maybe a sensor on seats to detect when someone sits down (Brumitt et al. 2000). Often, proximity information is enough and requires no calibration step (Krumm and Hinckley 2004). For our location-based services, we note that in some cases, it is important to know who is at a given location, but in many cases, this information is not necessary. We place various simple sensors throughout the environment in locations that we think will provide the information needed to support the specific location-based services we currently have in mind. The solution we come up with is quite specialized.

It is unclear which approach is superior. The geometric model-based approach depends heavily on the performance of the person-tracking system. When it fails, it may return no useful position information, even when the given circumstances do not require precise position information, and even while the sensor and the underlying signal-processing algorithms may produce relevant intermediate results that could be used at a subsequent level of processing. The assumption of device independence ignores the fact that different sensing technologies typically have very different failure modes. The resulting application may perform poorly because of its design around modeling choices that follow the lowest common denominator. However, it could work very well and provide the utmost in flexibility if the person-tracking is very reliable.

The specialized approach of course suffers in all the ways that the general approach excels; each installation of the system may require significant innovation on the part of the designer. However, because it is tailored to the application, the system is more likely to gracefully handle sensing failure modes, and furthermore, is more likely to be less wasteful of resources, both physical and computational. Not surprisingly, the specialized approach is likely to exhibit better performance for a given application than the generalized approach.

The consideration of generalized versus specialized designs is a common engineering problem that is especially relevant in the realm of sensor and recognition-based systems. As our example illustrates, the two approaches may demand completely different sensors, representations, and modeling choices.

7.5.7 Traditional versus Nontraditional Interfaces

A question that naturally arises in the application of sensing and recognition to interactive systems is whether the design should emulate, augment, or completely replace the interfaces we already have.

It is probably not surprising that so many interactive sensor-based systems emulate the mouse. After all, once this functionality is achieved, the system is now relevant to the vast majority of the world's software. It is interesting to note, however, the degree to which the original development of the GUI hinged on the development of the mouse itself, and how to this day, the mouse is still the favored input device. This suggests that unless the new interactive system offers significant new functionality over the mouse, it will not be adopted, and that instead of forcing the new techniques on today's interfaces, designers should think about what changes in the interface are implied by new sensing systems.

Another approach is to augment or complement today's interfaces. For example, it is relatively simple to add an accelerometer to a mobile device that allows the user to position the cursor or scroll by tilting the device, and it is not hard to imagine how users could easily pick up this interaction (Hinckley et al. 2000).

Still another approach is to completely reinvent the interface. The risks are that the user is required to learn a completely new way of working, the designer is faced with developing the entire system, and the utility of the system will inevitably be compared with more established techniques. Again, unless there is a significant improvement to be had, users are unlikely to adopt the new approach.

Many of these issues are illustrated by recent research into interactive table systems. In its simplest form, an interactive table system might be no more than a large flat-panel touch screen turned horizontal to function as a table surface.

This simple change in orientation enables new applications that exploit the collaborative nature of multiple users gathering around a table (Shen et al. 2004). Furthermore, sensing techniques have been developed to support multiple points of contact on the surface, and to support bimanual interaction, or the multiple hands of multiple users. Distinct from wall displays, tables are able to hold objects, and accordingly, some interactive tables use sensing techniques to recognize objects placed upon them. This ability can be used to support various tangible user interface scenarios. For example, a puck placed on the table can enable a feature of the application; the user might rotate the puck to adjust an associated parameter, or it might call up a group of photos (Fitzmaurice, Ishii, and Buxton 1995; Ullmer, Ishii, and Glas 1998; see also Ishii and Ullmer, Chapter 21, this volume).

Clearly, in order to exploit the unique affordances and sensing capabilities of an interactive table, we must be willing to let go of many of today's GUI interfaces. For example, the multiple-touch and multiple-user aspect cannot be supported by the single-focus model of the GUI, and even the assumption that the display has a natural reading orientation (an "up" direction) may no longer be valid (Kruger et al. 2004; Shen, Lesh, and Vernier 2003). Finally, today's GUI model has no role for tangible user interfaces (UIs).

The designers of interactive table systems tend to follow a few guiding principles to innovate without alienating the user. First, there is a desire to leave the overburdened widgets of the modern GUI behind and instead rely on more of a direct manipulation style of interaction. For example, to rotate a virtual object on the screen, it may suffice to place two fingers anywhere on the object and move one finger about the other, much in the same way that one might rotate a piece of paper sitting on a desk. By using the multiple touch capability, we avoid the clumsy widget-heavy and mode-heavy rotation techniques typical of many drawing packages. The addition of pucks and other tangle UIs extends this approach; for example, while the puck is on the table, it may behave as knob (Patten, Ishii, and Pangaro 2001). Second, there is a trend to add widget-based interaction back into the direct manipulation framework, perhaps mainly to address the need to perform a variety of actions on the digital representation of the object. For example, e-mailing, printing, and contrast and color adjustment are just a few things that the user might want to do with their photos; these operations are outside of the scope of direct manipulation. These widget-based interactions can draw upon the advantages a multiple touch table interface provides. For example, putting two fingers down near an object may trigger a scrollbar-like widget in which bringing the two fingers closer or further apart adjusts a parameter (Wu and Balakrishnan 2003).

7.5.8 Evaluating Novel Interfaces

Determining whether the novel sensing and interaction model actually works is the domain of usability testing. No research into sensor and recognition-based input is complete without an evaluation showing its effectiveness. Unfortunately, the technical difficulty of getting such systems to work at all often leaves little time to quantify performance on a standard task. Furthermore, when the new work is in a preliminary state, it may not be instructive to compare the new technique against one that has had decades to evolve.

Many times, user study subjects are often so impressed by the "magic" of sensor-based systems that the sheer novelty of the interface can skew study results. Subjective surveys are likely to show bias in favor of the novel design (Nielsen 1994). This is a very difficult problem without an easy solution, particularly in the case of tangible UIs, tabletop interactive systems, and perceptual user interface systems. Longitudinal studies, which involve repeated sessions spread out over multiple days, can be used to minimize this effect, but such studies are expensive and time-consuming. As a result, many interactive sensing-based systems come with few convincing quantitative user studies that prove their utility. In the case of a system that uses tracking or detection to select an object, Fitts's Law studies can be an effective technique to compare pointing performance across very different systems (MacKenzie 1992).

Of course, often the point of the work is not to show a decrease in task completion time, a reduction in errors, or other more conventional metrics from the field of HCI. Rather, the goal may be to highlight completely new ways of conceptualizing the relationship between users and their machines, and to demonstrate that such innovations are technically feasible. There is often more of an emphasis on invention and design than on evaluation. The novelty that can sabotage a user study may even be a desirable effect, compelling users to engaging experiences they may not have otherwise had. Furthermore, traditional evaluation methods seem at odds with the goals of surprising, delighting, and entertaining the user. The field has only recently begun to recognize the need to develop ways to objectively evaluate interactive systems along these dimensions that are basic to the quality of life (Blythe et al. 2004; Norman 2003).

A fantastic example where simple sensing techniques were used to great effect to surprise and delight the user is the PingPongPlus system (Ishii et al. 1999). In this research prototype, the usual ping-pong table was augmented with sound effects and a top–down video projection onto the table surface, and electronics to sense where the ping-pong ball hits the surface during game play (see Figure 7.5). This sensing system used eight microphones mounted under the table, and custom electronics to triangulate where the ball hit the surface. The sound effects and video presentation reacted to each strike of the ball, in ways calculated to amuse the players and augment the game play in dramatic ways. For example, in one mode, the ball produces ripples on the table, while in another, thunderstorm audio and video effects build up as the length of the volley increases.

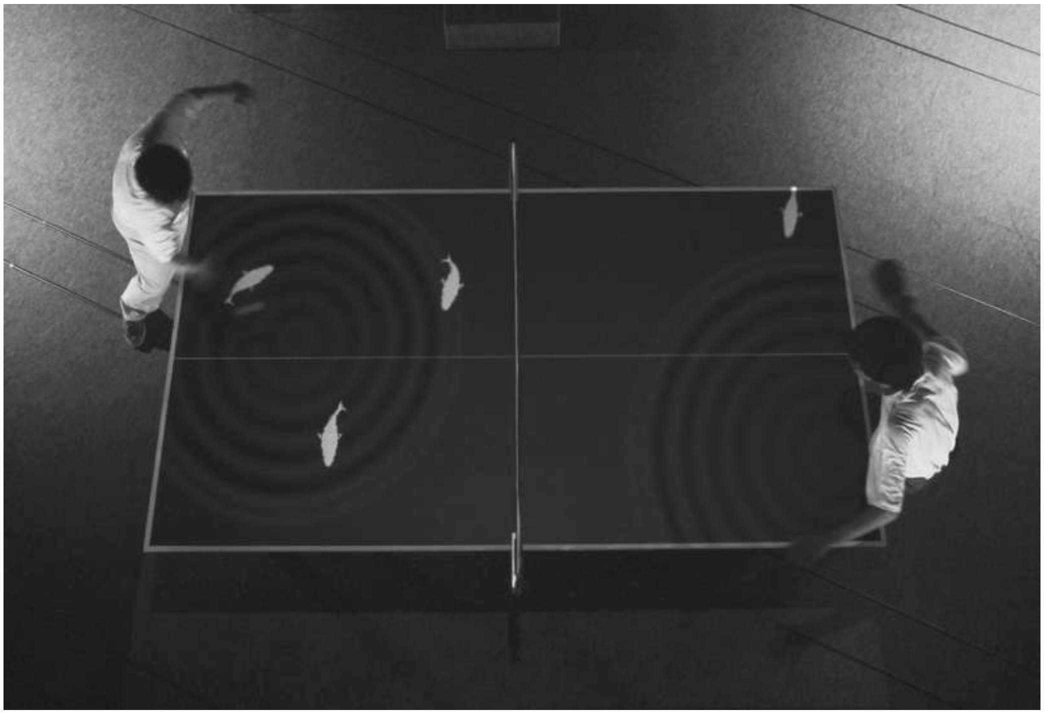


FIGURE 7.5 The PingPongPlus system uses a series of microphones to triangulate where the ping-pong ball strikes the surface of the table. This position information is used to drive a variety of interactive graphics displayed on the table by an overhead projector. (Courtesy of Tangible Media Group, MIT Media Laboratory, © 2006.)

7.6 CONCLUSION

In this chapter, we explored a variety of sensing technologies available today, outlined a number of signal-processing techniques common to using these sensing technologies in sensor and recognition-based input for interactive systems, and discussed further issues related to designing such systems.

Although much of the discussion highlighted difficulties in designing systems that rely on sensors, the future of interactive sensing systems is bright. Advances in MEMS and nanotechnology will continue to drive innovations in the sensors themselves, while the relentless increase in commodity CPU power and storage capabilities will continue to enable more sophisticated modeling techniques used in interpreting sensor outputs.

Another powerful driver in the development of sensing-based interactive systems is the growth of the computing form factor beyond the traditional desktop computer. The proliferation of cell phones, personal digital assistants, portable gaming devices, music players, tablet PCs, and living room-centric PCs shows a trend toward the vision of ubiquitous computing, in which computing is situated throughout our environment and daily life. Individual computing devices will be tailored in deep ways to the task at hand, away from the "one size fits all" mentality of desktop computing. The use of sensors and recognition techniques will play an important role in enabling this diversity, and will naturally support and demand a variety of interaction styles. As interactive systems become more tailored to a given activity, the opportunity to leverage the techniques described in this chapter increases, in turn enabling the application of the sensors themselves. Such a virtuous cycle may speed the development and adoption of sensing-based systems in ways that are hard to imagine today.

ACKNOWLEDGMENTS

The XWand and Orb prototypes were developed with the assistance of Mike Sinclair and Steven Bathiche. Thanks to Ken Hinckley, John Krumm, Steven Bathiche, and Ewa Davison for comments.

REFERENCES

Abowd, G. A., C. G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton. 1997. Cyberguide: A mobile context-aware tour guide. *ACM Wirel Netw* 3:421–33.

Abowd, G. A., A. Battestini, and T. O'Connell. 2003. *The Location-Service: A Framework for Handling Multiple Location Sensing—Technologies* (GVU Technical Report GIT-GVU-03-07). Atlanta, Georgia: Georgia Institute of Technology.

Bao, L., and S. Intille. 2004. Activity recognition from user-annotated acceleration data. In *Proceedings of the Second International Conference in Pervasive Computing, 3001/2004*, 1–17.

Baudel, T., and M. Beaudouin-Lafon. 1993. Charade: Remote control of objects using free-hand gestures. *Commun ACM* 36(7):28–35.

Baxter, L. 1996. *Capacitive Sensors: Design and Applications*. Hoboken, NJ: Wiley-IEEE Press.

Bellotti, V., M. Back, W. K. Edwards, R. E. Grinter, A. Henderson, and C. Lopes. 2002. Making sense of sensing systems: Five questions for designers and researchers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 415–22.

Beymer, D., and M. Flickner. 2003. Eye gaze tracking using an active stereo head. In *IEEE Conference on Computer Vision and Pattern Recognition*, 451.

Beymer, D., and T. Poggio. 1996. Image representations for visual learning. *Science* 272:1905–09.

Bilmes, J. A., X. Li, J. Malkin, K. Kilanski, R. Wright, K. Kirchoff, et al. 2005. The vocal joystick: A voice-based human-computer interface for individuals with motor impairments. In *Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, 995–1002.

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford University Press.

Blythe, M. A., K. Overbeeke, A. F. Monk, and P. C. Wright, ed. 2004. *Funology: From Usability to Enjoyment. Human-Computer Inter-action Series*. Dordrecht, The Netherlands: Kluwer.

Bobick, A., S. Intille, J. Davis, F. Baird, C. Pinhanez, L. Campbell, et al. 1999. The kidsroom. A perceptually-based interactive and immersive story environment. *Pressure Teleoperators Virtual Environ* 8(4):367–91.

Bobick, A., S. Intille, J. Davis, F. Baird, C. Pinhanez, L. Campbell, Y. Ivanov, A. Schutte, and A. Wilson. 1999. The Kidsroom: A perceptually-based interactive and immersive story environment. *Presence: Teleoperators and Virtual Environ* 8(4):367–91.

Boyd, J., and J. Little. 2005. Biometric gait recognition. In *Advanced Studies in Biometrics: Summer School on Biometrics, Alghero, Italy, June 2–6, 2003*, ed. M. Tistarelli, J. Bigun, and E. Grosso, 19–42. Revised Selected Lectures and Papers, Vol. 3161/2005. New York: Springer.

Brashear, H., T. Starner, P. Lukowicz, and H. Junker. 2003. Using multiple sensors for mobile sign language recognition. In *Proceedings of the 5th International Symposium on Wearable Computing*, 45–52.

Breiman, L., J. H. Freidman, R. A. Olsen, and C. J. Stone. 1984. Classification and regression trees. Boca Raton, FL: Chapman & Hall/CRC.

Brumitt, B., and J. Cadiz. 2001. Let there be light: Examining interfaces for homes of the future. In *Proceedings of Interact '01*, 375–82.

Brumitt, B. L., B. Meyers, J. Krumm, A. Kern, and S. Shafer. 2000. EasyLiving: Technologies for intelligent environments. In *Proceedings of the Handheld and Ubiquitous Computing, 2nd International Symposium*, 12–27.

Caruso, M. J. 1997. Applications of magnetoresistive sensors in navigation systems. *Sensors* Actuators 15–21.

Choudhury, T., and A. Pentland. 2003. Sensing and modeling human networks using the Sociometer. In *Proceeding of the International Conference on Wearable Computing*, 216–22.

Cristianini, N., and J. Shawe-Taylor. 2000. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, UK: Cambridge University Press.

Crowley, J. L., and F. Berard. 1997. Multi-modal tracking of faces for video communications. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 640–5.

Curran, E., and M. J. Stokes. 2003. Learning to control brain activity: A review of the production and control of EEG components for driving brain-computer interface (BCI) systems. *Brain Cogn* 51:326–36.

Davies, N., K. Ceverst, K. Mitchell, and A. Efrat. 2001. Using and determining location in a context-sensitive tour guide. *IEEE Comput* 34(8):35–41.

Dey, A. K., G. Kortuem, D. R. Morse, and A. Schmidt, ed. 2001. Special issue on situated interaction and context-aware computing. *Pers Ubiquitous Comput* 5(1):1–3.

Dietz, P., and D. Leigh. 2001. DiamondTouch: A multi-user touch technology. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, 219–26.

Fails, J. A., and D. Olsen. 2002. Light widgets: Interacting in everyday spaces. In *Proceedings of the 7th International Conference on Intelligent User Interfaces*, 63–9.

Feng, J., C.-M. Karat, and A. Sears. 2005. How productivity improves in hands-free continuous dictation tasks: lessons learned from a longitudinal study. *Interact Comput* 17(3):265–89.

Fischler, M. A., and R. C. Bolles. 1981. Random sample consensus for model fitting with applications to image analysis and automated cartography. *Commun ACM* 24:381–95.

Fitzmaurice, G. W., H. Ishii, and W. Buxton. 1995. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 442–9.

Fodor, I. K. 2002. *A Survey of Dimension Reduction Techniques*. Livermore, CA: Center for Applied Scientific Computing, Lawrence Livermore National Laboratory.

Forsyth, D. A., and Ponce, J. 2002. Computer vision: A modern approach. Upper Saddle River, NJ: Prentice Hall.

Fraden, J. 2003. *Handbook of Modern Sensors: Physics, Designs, and Applications*. New York: Springer.

Galyean, T. 1995. *Narrative Guidance of Interactivity*. Unpublished doctoral thesis, Media Laboratory. Cambridge: Massachusetts Institute of Technology.

Garfinkel, S., and B. Rosenberg, ed. 2005. *RFID: Applications, Security, and Privacy*. Boston, MA: Addison-Wesley.

Greenberg, S., and C. Fitchett. 2001. Phidgets: Easy development of physical interfaces through physical widgets. In *Proceedings of the UIST 2001 14th Annual ACM Symposium on User Interface Software*, 209–18.

Guiard, Y. 1987. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *J Mot Behav* 19:486–517.

Han, J. Y. 2005. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, 115–8.

Hariharan, R., J. Krumm, and E. Horvitz. 2005. Web-enhanced GPS. In *Location- and Context-Awareness: First international Workshop, LoCA 2005, Oberpfaffenhofen, Germany, May 12–13, 2005, Proceedings*, ed. T. Strang and C. Linhoff-Popien, 95–104. New York: Springer.

Haro, A., M. Flicker, and I. Essa. 2000. Detecting and tracking eyes by using their physiological properties, dynamics and appearance. In *Proceedings of the IEEE CVPR 2000*, 163–8.

Heckerman, D., D. Geiger, and D. M. Chickering. 1994. Learning Bayesian networks: The combination of knowledge and statistical data. In *10th Conference on Uncertainty in Artificial Intelligence*, 293–301.

Hinckley, K., R. Pausch, D. Proffitt, and N. Kassel. 1998. Two-handed virtual manipulation. *ACM Trans Comput Hum Interact (TOCHI)* 5(3):260–302.

Hinckley, K., J. Pierce, M. Sinclair, and E. Horvitz. 2000. Sensing techniques for mobile interaction. In *Proceedings of the ACM UIST 2000 Symposium on User Interface Software and Technology*, 91–100.

Hinckley, K., and M. Sinclair. 1999. Touch-sensing input devices. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (SIGGCHI '99)*, 223–30.

Hjelm, S. I., and C. Browall. 2000. Brainball—using brain activity for cool competition. Paper presented at NordiCHI 2000, Stockholm, Sweden.

Hoel, P. G., S. C. Port, and C. J. Stone. 1971. *Introduction to Probability Theory*. Boston, MA: Houghton Mifflin.

Horn, B. K. P. 1986. *Robot Vision*. Cambridge, MA: MIT Press.

Horvitz, E. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 159–66.

Horvitz, E., J. Breese, D. Heckerman, D. Hovel, and K. Rommelse. 1998. The Lumiere project: Baysian user modeling for inferring the goals and needs of software users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 256–65.

Huber, P. J. 1981. *Robust Statistics*. New York: John Wiley & Sons.

Ishii, H., C. Wisneski, J. Orbanes, B. Chun, J. Paradiso. 1999. PingPongPlus: Design of an athletic-tangible interface for computer-supported cooperative play. In *Proceedings of the Conference on Human Factors in Computing Systems (SIGCHI)*, 327–8.

Jacob, R. J. K. 1993. Eye-gaze computer interfaces: what you look at is what you get. *IEEE Comput* 26(7):65–7.

Jensen, F. V. 2001. *Bayesian Networks and Decision Graphs*. New York: Springer.

Johansson, G. 1973. Visual perception of biological motion and a model for its analysis. *Percept Psychophys* 14:201–11.

Jojic, N., B. Brumitt, B. Meyers, S. Harris, and T. Huang. 2000. Detection and estimation of pointing gestures in dense disparity maps. In *Proceedings of the Fourth International Conference on Automatic Face and Gesture Recognition*, 468–75.

Jordan, M., ed. 1999. *Learning in Graphical Models*. Cambridge, MA: MIT Press.

Junqua, J. C. 1993. The Lombard reflex and its role on human listeners and automatic speech recognizers. *J Acoust Soc Am* 93(1):510–24.

Kadie, C. M., D. Hovel, and E. Horvitz. 2001. *MSBNx: A Component-Centric Toolkit for Modeling and Inference with Bayesian Networks* (Microsoft Research Technical Report MSR-TR-2001-67). Redmond, WA: Microsoft Corporation.

Kahn, J. M., R. H. Katz, and K. S. J. Pister. 2000. Emerging challenges: Mobile networking for "smart dust." *J Commun Netw* 2(3):188–96.

Karat, C.-M., C. Halverson, J. Karat, and D. Horn. 1999. Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Proceedings of CHI '99*, 568–75.

Kato, H., M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. 2000. Virtual object manipulation on a tabletop AR environment. In *Proceedings of ISAR 2000*, 111–9.

Kiern, Z. A., and J. I. Aunon 1990. A new mode of communication between man and his surroundings. *IEEE Trans Biomed Eng* 37(12):1209–14.

Klein, J., Y. Moon, and R. W. Picard. 2002. This computer responds to user frustration: Theory, design and results. *Interact Comput* 14(2002):119–40.

Kovacs, G. T. 1998. *Micromachined Transducers Sourcebook*. New York: McGraw-Hill.

Kruger, R., S. Carpendale, S. D. Scott, and S. Greenberg. 2004. Roles of orientation in tabletop collaboration: Comprehension, coordination and communication. *Comput Support Coop Work* 13(5–6):501–37.

Krumm, J., and K. Hinckley. 2004. The NearMe wireless proximity server. In *Sixth International Conference on Ubiquitous Computing (Ubicomp 2004)*, 283–300.

Krumm, J., and E. Horvitz. 2004. Locadio: Inferring motion and location from Wi-Fi Signal Strengths. In *First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (Mobiquitous 2004)*, 4–13.

LaMarca, A., Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, et al. 2005. Place lab: Device positioning using radio beacons in the wild. *Pervasive Comput* 3468/2005:116–33.

LaViola, J., and R. Zeleznik. 2004. Mathpad2: A system for the creation and exploration of mathematical sketches. In *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2004)*, 432–40.

Lee, S. W., and K. Mase. 2002. Activity and location recognition using wearable sensors. *IEEE Pervasive Comput* 1(3):24–32.

Lester, J., T. Choudhury, and G. Borriello. 2006. A practical approach to recognizing physical activities. In *Proceedings Pervasive Computing 2006*, 1–16.

Letchner, J., D. Fox, and A. LaMarca. 2005. Large-scale localization from wireless signal strength. AAAI, 15–20.

Li, S. Z., and A. K. Jain. 2005. *Handbook of Face Recognition*. New York: Springer.

Lowe, D. 2004. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* 60(2):91–110.

MacKenzie, I. S. 1992. Fitts' Law as research and design tool in human computer interaction. *Hum Comput Interact* 7:91–139.

MacKenzie, I. S., and C. Ware. 1993. Lag as a determinant of human performance in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 488–93.

Maes, P., T. Darrell, B. Blumberg, and A. Pentland. 1995. The ALIVE system: Full-body interaction with autonomous agents. In *Proceedings of the Computer Animation*, 11.

Matsushita, N., and J. Rekimoto. 1997. HoloWall: Designing a finger, hand, body and object sensitive wall. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 209–10.

McNeill, D. 1992. *What Gestures Reveal About Thought*. Chicago: The University of Chicago Press.

Mitchell, T. M. 1997. *Machine Learning*. Boston: McGraw-Hill.

Mozer, M. C. 2005. Lessons from an adaptive house. In *Smart Environments: Technologies, Protocols and Applications*, ed. D. Cook and R. Das, 273–94. Hoboken, NJ: J. Wiley & Sons.

Munguia Tapia, E., S. S. Intille, and K. Larson. 2004. Activity recognition in the home setting using simple and ubiquitous sensors, in *Proceedings of PERVASIVE 2004*, vol. LNCS 3001, eds. A. Ferscha and F. Mattern, 158–75. Berlin Heidelberg: Springer-Verlag.

Nachman, L., R. Kling, J. Huang, and V. Hummel. 2005. The Intel mote platform: A Bluetooth-based sensor network for industrial monitoring. In *Fourth International Symposium on Information Processing in Sensor Networks*, 437–42.

Neal, R., and G. Hinton. 1999. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, ed. M. I. Jordan, 355–68. Cambridge, MA: MIT Press.

Nielsen, J. 1994. *Usability Engineering*. San Francisco: Morgan-Kaufmann.

Norman, D. A. 2003. *Emotional Design: Why we Love (or hate) Everyday Things*. New York: Basic Books.

Oliver, N., and E. Horvitz. 2005. Selective perception policies for guiding sensing and computation in multimodal systems: A comparative analysis. *Comput Vis Image Underst* 100(1–2):198–224.

Oviatt, S. L. 2002. Breaking the robustness barrier: Recent progress on the design of robust multimodal systems. In *Advances in Computers*, ed. M. Zelkowtiz, Vol. 56, 305–41. London: Academic Press.

Patten, J., H. Ishii, and G. Pangaro. 2001. Sensetable: A wireless object tracking platform for tangible user interfaces. In *Proceedings of the Conference on Human Factors in Computing Systems (SIGCHI)*, 253–60.

Pentland, A. 2004. Social dynamics: Signals and behavior. In *Third International Conference on Development and Learning*, 263–7.

Pentland, A., B. Moghaddam, and T. Starner. 1994. View-based and modular eigenspaces for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 84–91.

Picard, R. W. 2000. *Affective Computing*. Cambridge, MA: The MIT Press.

Pickering, J. 1986. Touch-sensitive screens: The technologies and their application. *Int J Man Mach Stud* 25(3):249–69.

Platt, J. 1999. Using analytic QP and sparseness to speed training of support vector machines. *Adv Neural Inf Process Syst* 11:557–63.

Rabiner, L. R. 1989. A tutorial in hidden Markov models and selected applications in speech recognition. *Proc IEEE* 77(2):257–386.

Rabiner, L., and B. Juang. 1993. *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice Hall.

Rekimoto, J. 2002. SmartSkin: An infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 113–20.

Rekimoto, J., and Y. Ayatsuka. 2000. CyberCode: Designing augmented reality environments with visual tags. In *Proceedings of the Designing Augmented Reality Environments (DARE 2000)*, 1–10.

Reynolds, C. 2001. *The Sensing and Measurement of Frustration with Computers*. Unpublished master's thesis. Cambridge: Media Laboratory, Massachusetts Institute of Technology.

Rubine, D. 1991. Specifying gestures by example. *Comput Graph* 25(4):329–37.

Rui, Y., and D. Florencio. 2003. New direct approaches to robust sound source localization. In *Proceedings of IEEE International Conference on Multimedia Expo*, 737–40.

Schapire, R. E. 2003. The boosting approach to machine learning: An overview. In *Nonlinear Estimation and Classification*, ed. D. D. Denison, M. H. Hansen, C. Holmes, B. Mallick, and B. Yu. New York: Springer.

Schmidt, A. 2000. Implicit human interaction through context. *Pers Technol* 4(2&3):191–9.

Schmidt, A., M. Beigl, and H. W. Gellersen. 1999. There is more to context than location. *Comput Graph J* 23(6):893–902.

Sears, A., C. Plaisant, and B. Shneiderman. 1993. A new era for high precision touchscreens. In *Advances in Human-Computer Interaction*, Vol. 3, eds. H. Rex Hartson and D. Hix, 1–33. Norwood, NJ: Ablex Publishing Corp.

Shell, J., R. Vertegaal, D. Cheng, A. W. Skaburskis, C. Sohn, A. J. Stewart, et al. 2004. ECSGlasses and EyePliances: Using attention to open sociable windows of interaction. In *Proceedings of the ACM Eye Tracking Research and Applications Symposium*, 93–100.

Shen, C., N. Lesh, and F. Vernier. 2003. Personal digital historian: Story sharing around the table. *ACM Interact* 10(2):15–22.

Shen, C., F. D. Vernier, C. Forlines, and M. Ringel. 2004. DiamondSpin: An extensible toolkit for around-the-table interaction. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (SIGCHI)*, 167–74.

Shumin, Z., and V. Bellotti. 2005. Introduction to sensing-based interaction. *ACM Trans Comput-Hum Interact* 12(1):1–2. DOI=10.1145/1057237.1057238. http://doi.acm.org/10.1145/1057237.1057238.

Smith, J. R. 1999. *Electric Field Imaging*. Unpublished doctoral dissertation. Cambridge: Media Laboratory, Massachusetts Institute of Technology.

Smith, A., H. Balakrishnan, M. Goraczko, and N. Priyantha. 2004. Tracking moving devices with the Cricket location system. In *Proceedings of the 2nd USENIX/ACM MOBISYS Conference*, 190–202.

Stork, D. G. 1998. *HAL's legacy: 2001's Computer as Dream and Reality*. Cambridge, MA: MIT Press.

Sugiura, A., and Y. Koseki. 1997. A user interface using fingerprint recognition—holding commands and data objects on fingers. In *Proceedings of the Symposium on User Interface Software and Technology*, 71–9.

Tan, H. Z., L. A. Slivovsky, and A. Pentland. 2001. A sensing chair using pressure distribution sensors. *IEEE/ASME Trans Mechatronics* 6(3):261–8.

Tashev, I., and H. S. Malvar. 2005. A new beamformer design algorithm for microphone arrays. In *Proceedings of International Conference of Acoustic, Speech and Signal Processing*, 101–04.

Tenneson, D., and S. Becker. 2005. ChemPad: Generating 3D modelcules from 2D sketches. SIGGRAPH 2005 Extended Abstracts.

Tobii Technology. 2005. http://www.tobii.com (accessed February 10, 2007).

Tomasi, C., A. Rafii, and I. Torunoglu. 2003. Full-size projection keyboard for handheld devices. *Commun. ACM* 46(7):70–5.

Ullmer, B., H. Ishii, and D. Glas. 1998. MediaBlocks: Physical containers, transports, and controls for online media. *Comput Graph* 32:379–86.

Vogel, D., and R. Balakrishnan. 2004. Interactive public ambient displays: Transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2004)*, 137–46.

Vogel, D., and R. Balakrishnan. 2005. Distant freehand pointing and clicking on very large high resolution displays. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 33–42.

Vranish, J. M., R. L. McConnell, and S. Mahalingam. 1991. "Capaciflector" collision avoidance sensors for robots. *Robot Res NASA Goddard Space Flight Center* 17(3):173–9.

Want, R., K. P. Fishkin, A. Gujar, and B. L. Harrison. 1999. Bridging physical and virtual worlds with electronic tags. In *SIGCHI '99*, 370–7.

Want, R., A. Hopper, V. Falcao, and J. Gibbons. 1992. The active badge location system. *ACM Transact Inf Syst (TOIS)* 10(1):91–102.

Ward, A., A. Jones, and A. Hopper. 1997. A new location technique for the active office. *IEEE Pers Commun* 4(5):42–7.

Ware, C. 1990. Using hand position for virtual object placement. *Vis Comput* 6(5):245–53.

Ware, C., and D. R. Jessome. 1988. Using the Bat: A six-dimensional mouse for object placement. In *Proceedings of the IEEE Computer Graphics and Applications*, 65–70.

Wayman, J., A. Jain, D. Maltoni, and D. Maio, ed. 2004. *Biometric Systems: Technology, Design and Performance Evaluation*. New York: Springer.

Welch, G., and G. Bishop. 2004. *An Introduction to the Kalman Filter*. (Dept. Computer Science Technical Report TR-95-041). Chapel Hill, NC: University of North Carolina at Chapel Hill.

Wellner, P. 1993. Interacting with paper on the DigitalDesk. *Commun ACM* 36(7):87–96.

Wilson, A. D. 2005. PlayAnywhere: A compact tabletop computer vision system. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software Technology*, 83–92.

Wilson, D. H., and C. G. Atkeson. 2005. Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors. In *Proceedings of Pervasive 2005*, 62.

Wilson, A. D., and H. Pham. 2003. Pointing in intelligent environments with the WorldCursor. In *Proceedings of Interact*, 495–502.

Wilson, A. D., and S. Shafer. 2003. XWand: UI for intelligent spaces. In *Proceedings of SIGCHI*, 545–52.

Wren, C., A. Azarbayejani, T. Darrell, and A. Pentland. 1995. Pfinder: Real-time tracking of the human body. In *Proceedings of SPIE Photonics East*, 89–98.

Wu, M., and R. Balakrishnan. 2003. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proceedings of the 16th Annual Symposium on User Interface Software and Technology*, 193–202.

Wu, Y., K. Toyama, and T. Huang. 2000. Wide-range, person- and illumination-insensitive head orientation estimation. In *Proceedings of the International Conference on Face and Gesture Recognition*, 183.

Zhai, S., C. Morimoto, and S. Ihde. 1999. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 246–53.

Zhang, Z., and J. Hershey. 2005. *Fusing Array Microphone and Stereo Vision for Improved Computer Interfaces* (Microsoft Research Technical Report MSR-TR-2005-174) Redmond, WA: Microsoft Corporation. Smart Technologies, Incorporated. http://www.smarttech.com (accessed February 10, 2007).

Zhao, W., R. Chellappa, R. J. Phillips, and A. Rosenfeld. 2003. Face recognition: A literature survey. *ACM Comput Surv* 35(4):399–458.

8 Visual Displays

Christopher M. Schlick, Carsten Winkelholz, Martina Ziefle, and Alexander Mertens

CONTENTS

8.1 INTRODUCTION

Since the dawn of humankind, people have been inventing and building tools to make life easier and more comfortable. One of mankind's first tools with a visual display was the sundial, built more than 3000 years ago in Babylon. Due to its basic physical principle though, one of its significant disadvantages was that it could not show the time during the night or when the weather was cloudy. Moreover, the first sundials were in public places, so people had to make the effort to go there to check the time. Later developments of clocks (also in public places) eliminated the disadvantages of the sundial. Church clocks could show the time in almost any type of weather and at any time of day or night, and they could also display the time acoustically and therefore bridge distances of up to a few miles. The following developments are well known—from the grandfather clock and the fob watch to the first analog and digital wristwatches to today's powerful wrist computers with high-resolution active-matrix organic light-emitting diode displays (AMOLED) (Vogel et al. 2008). These displays not only show the time and date ergonomically under different ambient light intensities, but also present complex information in full color with short response times. This means user interface designers can chart a person's precise location on earth using the Global Positioning System, visualize individual heart rate profiles, and even present high-dynamic range video images.

8.2 BASIC PRINCIPLES AND CONCEPTS

8.2.1 Image Generation

Visual displays rely on different physical principles to generate an image, including light emission, transmission, and reflection. Examples from noncomputer displays can be helpful to explain these basic principles and point out relevant advantages and limitations.

Writing on a piece of paper alters the reflective properties of the paper from a highly reflective white to a less reflective blue or black. Ambient light is needed to read what is written on the paper, but the contrast ratio between the text and the background remains the same in different intensity conditions. A similar principle holds true for reflectance-based displays such as classic reflective liquid crystal displays (LCDs) or electrophoretic displays (see Sections 8.4.3 and 8.4.8).

Examples of transmission include transparency film used with an overhead projector and slides used with a projector. Different parts of the transparency or slide transmit light of different wavelengths (i.e., color) with different intensity. Transmission is used as a basic principle in many electronic information displays, such as the popular thin film transistor (TFT) LCDs. It is also often combined with other physical principles. For instance, a simple front projection display (see Section 8.4.1) consists of a light source, a TFT LCD that forms the image by transmitting the rays of light through a matrix of liquid crystals picture elements, and finally a projection surface that reflects the light into the eye of the observer.

An example of emission would be a lighthouse. Its light can be easily seen at night, but it is barely visible in bright sunlight. Examples of emission-based displays are cathode ray tubes (CRTs) (see Section 8.4.2), electroluminescent displays (see Section 8.4.5) and cathodoluminescent displays (see Section 8.4.6). Like a lighthouse, these displays need to be brighter than the ambient light to be perceived properly.

8.2.2 Electronic Information Displays

The development of advanced display technologies began with CRT, which was invented in the nineteenth century, though the observation of a glow from the electronic excitation of gas in an evacuated tube may go back as early as the seventeenth century. The invention of the device itself is generally attributed to Karl Ferdinand Braun. The "Braun tube" reportedly first built in Strasbourg, Germany in 1897, used an evacuated tube, deflection plates, and a fluorescent material for the screen. It was probably the first use of an electronic information display in the natural sciences (Castellano 1992).

Over the last decade, flat-panel displays based on LCD technology have been becoming increasingly popular for human–computer interfaces, gradually replacing the CRT displays that have dominated the market since the 1960s. Yet increasing quality demands from consumers have also led to rapid developments in alternative technologies for electronic information displays, such as the above-mentioned AMOLED displays. Today, a large variety of display technologies are competing over image quality, screen size, costs, power consumption, 3D imaging, and novel information input features for mobile applications (see Section 8.7.3).

There are two basic methods for displaying information visually: digital and analog. A digital display uses a discrete spatial structure (typically a square or rectangular grid of picture elements) to display symbols such as characters and icons, whereas an analog system uses a spatially continuous instrument panel for information presentation. If an instant impression is required, analog displays often present information better. Many people glance quickly at their analog watch and know roughly what the time is or at the analog speed indicator of their automobile dashboard and know that they are driving too fast. Analog displays translate a value of a continuous variable into an angle or a distance. Analog displays used for control devices consist of a scale and an indicator. Either the scale or the indicator moves. There are a number of guidelines for designing analog displays (Woodson 1987; Baumann and Lanz 1998).

When accuracy is a critical issue, however, digital displays are preferred. Reading analog meters quickly and accurately requires time and cognitive skills, whereas writing down the value from a digital display is merely a case of copying down the digits. In cases where both accuracy and quick reckoning are required, hybrid displays are often used.

A computer is a digital device, so all symbols it displays are encoded into binary numbers and drawn as matrices of picture elements. Therefore, all commonly used electronic information displays are digital displays. Nevertheless, some application programs mimic indicator devices for their ergonomic advantages. If the spatial resolution and the response time of a digital display are sufficiently high and the covered color space is sufficiently large, there will be no significant differences in visual performance versus an analog display (see Section 8.3).

8.2.3 Display Segmentation

Standard PC monitors have a sufficiently high resolution to display a virtually unlimited set of characters, icons, and graphics. Conversely, a wide variety of displays with

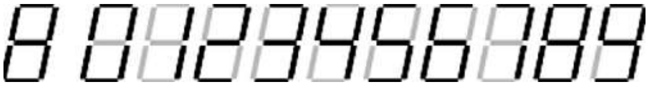


FIGURE 8.1 Seven-segment display.

lower spatial resolution can be found on other kinds of computerized technical devices such as low cost mobile phones or music players. Here, the set of displayable tokens is often far more restricted.

Basic display elements are binary, being either on or off. A ternary display element can be built from a two-color light-emitting diode (LED). To transmit more information, display elements with more states are needed. These could use different colors or luminance levels.

A common way to increase the amount of displayable information is by grouping a number of binary display elements into a unit. The classical seven-segment display (Figure 8.1) and many custom displays are examples of this approach.

If multiple display elements that have the same shape are arranged in matrix form and the elements are explicitly addressed to display multiple colors or different luminance levels, the basic concept of the common direct-view and projection displays can be derived. The shape of picture elements (pixels) in current electronic information displays is typically square or rectangular. For two-dimensional (2D) image data, the position of each pixel is addressed in horizontal rows and vertical columns called pixel matrices or pixel formats. In 3D space, volume elements are called voxels (volumetric pixels). Their positions are normally addressed in Cartesian space.

8.2.4 Display Dimensionality

Visual displays can be differentiated by means of the spatial dimensionality of the image generated. The simplest way to display information is to rely on only one spatial dimension. However, the notion of "1D display" is not strictly correct because human perception requires the picture elements to have two spatial dimensions; the secondary dimension of such displays does not provide information and is simply a function of the primary dimension. 1D displays represent a bare-minimum approach to presenting information, which is encoded by either discrete or continuous variables in 1D-state space. Still, a simple source of light such as a LED can convey a binary message of arbitrary length. Most current systems show the status of devices as "on" or "off" with LEDs. LEDs give off light radiation when biased in the forward direction. When displaying a continuous state variable in one dimension, an indicator bar value usually indicates the actual discretized value being measured or controlled. The volume level of a speaker or the temperature in a room can be indicated using a simple row of LEDs or an indicator bar on a linear scale.

To display more complex information that can only be shown using two independent spatial dimensions (length and width), 2D displays are used. This category of displays occupies the biggest segment in the market by far. 2D displays are suitable for displaying all kinds of visual information. Images are formed by activating pixels, each of these elements has a unique location (x,y) on a 2D plane, and a color scale or gray scale value can be assigned to each one. Although these displays have only two dimensions, depth perception is also achievable by using depth cues such as relative size, height relative to the horizon, interposition or occlusion, shadows and shading, spatial perspective, linear perspective, and texture gradients. Even more realistic three-dimensional (3D) scenes can be viewed on 2D screens by applying special viewing devices such as shutter glasses or polarizing glasses.

8.2.5 Human-Related Display Functionality

The wearable computing paradigm requires visual displays to be worn on the human body (Azuma 1997; Amft and Lukowicz 2009). A popular approach is to mount the display on a user's head. Thus the user can work hands-free and is always able to perceive information in his or her field of view. These displays are commonly referred to as head-mounted displays (HMDs). From a technological point of view, HMDs can roughly be divided into three main categories based on the way the image is provided to the user (Figure 8.2).

The first category, screen-based HMDs, comprises all HMDs whose picture elements are arranged in a spatially adjacent way. Ocular image forming displays use technologies such as LCD, digital mirror devices (DMDs), or organic light-emitting diodes (OLEDs). Most of the HMDs on the market today, however, are based on transmissive or reflective liquid crystal minidisplays (LCD) (Holzel 1999). Miniature LCDs are available at a relatively low price, provide suitable resolution (SXGA resolution of up to 1280 × 1024 pixels) and are lightweight (von Waldkirch 2004). With the retinal projection method, the image is projected directly onto the retina the same way a slide is projected onto a screen. Retinal projection displays are normally designed in the form of a Maxwellian-view optical system (Bass 1995), where the screen plane is optically conjugated to the retina and the illumination source is conjugated to the eye's pupil plane. Consequently, such displays can only be implemented on the basis of illuminated screens (like LCD and DMD) and not with self-emitting technologies like OLED and CRT (von Waldkirch 2004).

Scanning displays, where the image is scanned pixel-by-pixel directly onto a surface, are an alternative to screen-based displays. A retinal scanning display (RSD), also referred to as virtual retinal displays (VRDs), is the most important in this category. VRD technology was first proposed in 1992 by Sony. Since 1992, researchers at the Human Interface Technology Lab (Washington, DC) have been developing this technology to obtain a commercial product. In 2003, they presented the first commercial VRD (called Nomad) in collaboration with the U.S. company Microvision. This technology scans an image directly onto a viewer's retina using low-power red, green, and blue light sources such as lasers or LEDs (see Section 8.4.7). The VRD system has superior brightness and contrast compared with LCDs and CRTs because it typically uses spectrally pure lasers as a

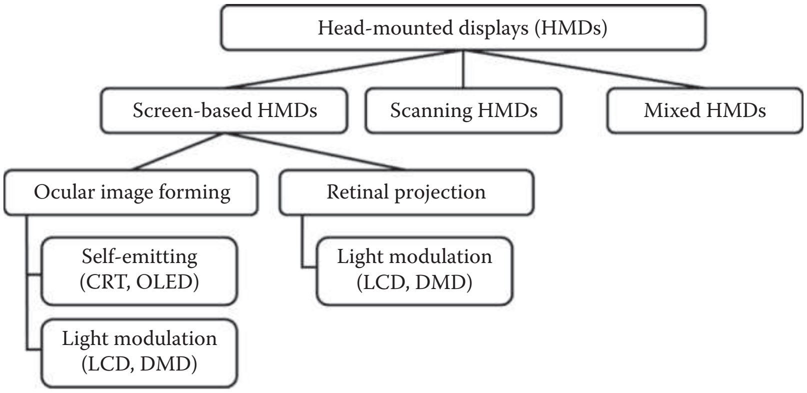

Head-mounted displays (HMDs)
Screen-based HMDs
Scanning HMDs
Mixed HMDs
Ocular image forming
Retinal projection
Self-emitting (CRT, OLED)
Light modulation (LCD, DMD)
Light modulation (LCD, DMD)

FIGURE 8.2 Systematics of head-mounted displays. (Adapted from Von Waldkirch, M. 2004. Unpublished doctoral thesis, Zurich, Switzerland: Swiss Federal Institute of Technology.)

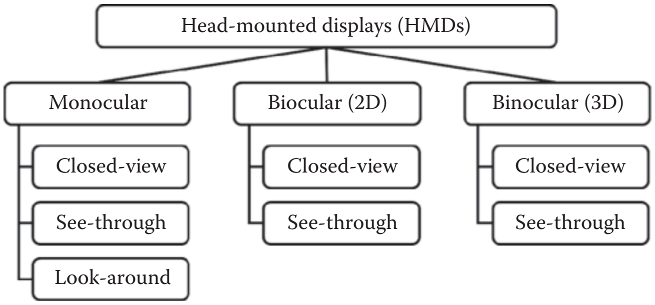


FIGURE 8.3 Head-mounted displays categorized according to visual perception. (According to Von Waldkirch, M. 2004. Unpublished doctoral thesis. Zurich, Switzerland: Swiss Federal Institute of Technology.)

light source (Stanney and Zyda 2002; Urey, Wine, and Lewis 2002). Recent developments use microelectromechanical systems scanner technology to present an ultrawide image of up to an estimated 100° horizontal by 90° vertical field of view at the pupil of each eye (Hilton 2008). Finally, there is the option of combining a scanning technology and a screen-based system (Fruehauf et al. 2000; Lippert 2006).

Besides technological categorization, HMDs can be classified according to visual perception (Figure 8.3, von Waldkirch 2004). Here, HMDs are usually divided into monocular, biocular, and binocular displays. In addition, all these types can provide the image in closed-view or see-through mode, and the monocular also in look-around mode.

See-through displays are usually problematic because visual information appears a specific distance away from the observer within the natural environment. To perceive the displayed information clearly, a user's eyes need to accommodate to that distance. This can be a problem if a user needs to attend to objects within the natural environment and to objects on the display simultaneously, provided the virtual distance to the displayed information and the distance to the real objects are not equal. According to von Waldkirch (2004), images displayed by different technologies have a different depth of focus. In an ideal case, the displayed image would be accommodation-insensitive. However, candidate technologies promising to increase depth of focus have a drawback: The sharpness of the image also in focus decreases (Winkelholz 2008).

Monocular displays have a single display source and provide the image to one eye only. Compared with binocular HMDs, monocular displays are lighter and cheaper. However, in a monocular HMD, the view seen by the eye not blocked by a screen may produce binocular rivalry with the image seen through the HMD. Biocular HMDs have two displays with separate screens and optical paths, enabling both eyes to see the same images simultaneously. Users only perceive a 2D image, like when using a computer screen. Binocular displays allow stereoscopic viewing with 3D depth perception. To produce the stereoscopic view, two spatially slightly incongruent images are presented to the left and to the right eye (binocular parallax, see Section 8.3.7).

Popular application areas for HMDs include defense, security, and technical services, where they can provide stereoscopic views of 3D CAD models for maintenance, overhaul, and repair of complex mechanical systems. An HMD can enable technicians to effectively "see through" a metal housing to view critical safety components such as valves (Schlick et al. 2009).

In addition to HMDs, electronic information displays that users hold in their hand must also be considered. Smartphones are one area where these types of displays are used. In recent years, mobile phones and personal digital assistants have merged into smartphones to provide integrated functionalities like calling, keeping an agenda, sending and receiving messages, playing games, music, and video, and surfing the Internet in one device (Wiley et al. 2008). For the latter applications especially, sufficient screen size, display resolution, and luminance are important requirements to ensure usability under different ambient light conditions and for users with different perceptual abilities (see Section 8.6.3). TFT LCD technology (see Section 8.4.3) and more recently AMOLED technology (see Section 8.4.5) are the primary display types used in handheld devices. An additional important requirement is that the electronic information display can be used as an input device by touching the plastic or glass cover of the screen (see Section 8.4.9). Satisfying all functional and ergonomic requirements is very challenging because displays often consume more than 50% of the total power available and therefore cannot simply be enlarged (Zehner 2008).

Advances in electronics have recently brought affordable, truly “wearable” visual displays to the consumer market. Researchers incorporated LEDs into textiles and called them “photonic textiles.” A flexible array of LEDs (see Section 8.4.5) and attached optical fibers are integrated into the textiles and connected to a battery power supply. Using the provided control unit, messages can be created and distributed wirelessly to several pieces of cloth (Graham-Rowe 2007). The location on the human body is critical for usability and acceptance because the information not only needs to be accessible for direct manipulation, located in a socially acceptable spot and comfortable to use, but it must also be transmitted effectively. Research in this field can roughly be divided into three areas (Tumler et al. 2008): (1) research that focuses on technological aspects, (2) research into areas of application such as military and industry, and (3) research concerned with ergonomic and medical aspects.

The cave automatic virtual environment (CAVE) is an “accessible” display in the truest meaning of the word. The CAVE creates an immersive virtual reality (VR) environment (see Section 8.7.1) around the user by aiming several projectors at the ambient walls. The walls of the CAVE are usually made up of rear-projection screens, and the floor is made of a down-projection screen. The first CAVE was developed in 1992 at the University of Illinois at Chicago in response to a challenge to demonstrate the manifold application areas of very large projection displays. Today CAVEs are used to design vehicle interiors and the layout of entire factory facilities, long before large investments in physical mock-ups or prototypes become necessary (DeFanti et al. 2009).

8.3 QUALITY CRITERIA

8.3.1 Colors

The retina of the human eye is covered with photoreceptors that transform the energy distribution of incoming light into neural activity. There are three main classes of photoreceptors: cones, rods, and photosensitive ganglion cells. Under normal-light conditions only three photoreceptors, known as cone cells, play a role in color perception. They differ in their spectral sensitivity. There are different types of cone cells for short (420–440 nm), medium (530–540 nm), and long (560–580 nm) wavelengths. Cone cells generate electrochemical impulses proportional to the inner product of their corresponding spectral sensitivities and the spectral distribution of the incoming light. Thus, in principle, color sensation can be described by three parameters, called tristimulus values. For an objective description of colors the Commission International de l’Eclairage (CIE) has defined three color matching functions, which can be thought of as the spectral sensitivity curves of three linear light detectors that yield the tristimulus values X, Y, and Z. However, these color-matching functions do not directly correspond to the spectral sensitivity curves of the cone types, and because it is not absolutely clear how the actual tristimulus values of the cone cells are processed, many different sets of color-matching functions can be specified to represent color sensations unambiguously. Figure 8.4 depicts the process of color perception from red, green, and blue (RGB) pixels. The RGB color model is an additive color model in which RGB light are mixed together in various ways to produce a broad range of colors. If any two of the RGB color channels are mixed in equal proportions, new colors are created: blue and green create cyan (bright, light blue); red and blue make magenta (a bright pink); and red and green create yellow. If all three colors are mixed together equally, the result is white light.

The color-matching functions of the CIE-XYZ system were defined to have some useful properties. For example, whenever equal amounts of XYZ values are combined, they match white light. Furthermore, the $\bar{y}(\lambda)$ color matching function was chosen to be exactly equal to the photopic luminous efficiency function $V(\lambda)$ for the “CIE standard photopic observer,” which describes the variation of perceived brightness with wavelength.

Because XYZ values scale with light intensity, they also encode the brightness of the color. To describe the actual color sensation, relative amounts denoted by lower case letters are used. Because x, y, and z are relative amounts, it is true that $x + y + z = 1$. Consequently, when x and y are

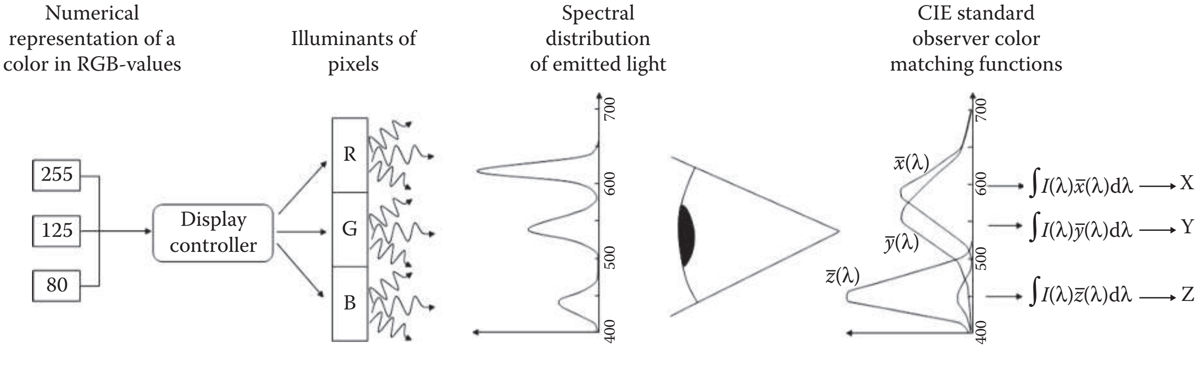

Numerical representation of a color in RGB-values
255
125
80
Display controller
Illuminants of pixels
R
G
B
Spectral distribution of emitted light
700
600
500
400
CIE standard observer color matching functions
$\bar{x}(\lambda)$
$\bar{y}(\lambda)$
$\bar{z}(\lambda)$
700
600
500
400
$\int I(\lambda)\bar{x}(\lambda)d\lambda \longrightarrow X$
$\int I(\lambda)\bar{y}(\lambda)d\lambda \longrightarrow Y$
$\int I(\lambda)\bar{z}(\lambda)d\lambda \longrightarrow Z$

FIGURE 8.4 Generation of color and modeling of perception of color by color matching functions of the standard XYZ system. The curves $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, and $\bar{z}(\lambda)$ represent the relative amounts of the X, Y, and Z primaries needed to match the color of the wavelength of light.

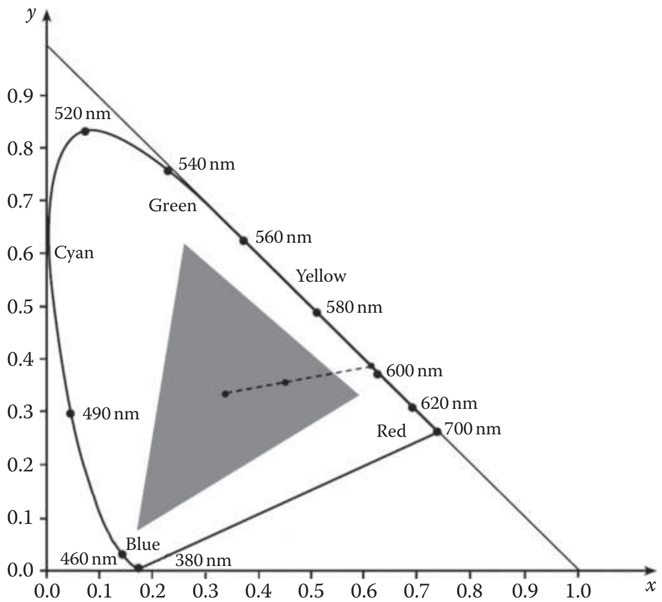


FIGURE 8.5 The Commission International de l'Eclairage chromaticity diagram. The curve, the locus of spectral colors, represents the *x*, *y*, and *z* values for all of the spectral colors between 380 nm and 770 nm. The triangle depicts the colors that can be presented by a display that uses the three primary colors at the corners of the triangle. The hue of any color within the curved line can be determined by drawing a line from pure white ($x = 0.33$ and $y = 0.33$) through the color of interest to the locus line. The saturation is given by the relative length of the dashed line between white and the color of interest and white and the intersection of this line with the locus. (Adapted from Schmidt, N. S. 2009. *Professionelle Videotechnik*. Berlin: Springer.)

known, *z* is known as well, because $z = 1 - (x + y)$. Therefore all colors can be represented within a *xy*-chromaticity diagram, which is shown in Figure 8.5. Pure white (i.e., achromatic light) is represented at the intersection of $x = 0.33$ and $y = 0.33$. The curved line, the locus of spectral colors, represents the *x* and *y* values for all of the spectral colors (see Figure 8.5). It is convenient to incorporate the Y-component for luminance into the description of a color and keep the *x*, *y* values as a description for chromaticity. This system of color characterization is referred to as the "xyY-system."

All visible colors can be encoded within the CIE-xyY system. Most luminous visual displays produce colors by using subpixels of the primary colors red, green, and blue. By controlling the luminance of the subpixels, different spectral distributions for each pixel can be generated, enabling different colors to be displayed for one pixel. Because of the (near) linearity of human color perception as expressed by Grassmann's Law, the resulting color in the XYZ system can be expressed by a linear combination of the intensities of the primary colors used. Because intensities can only be added and are bounded by an upper limit, a display is only able to produce colors that lie within a triangle in the *xy*- chromaticity diagram whose corners are defined by the primary colors used (see Figure 8.5). Computer systems encode the intensities of the primary colors of one pixel as RGB values. The representation of this information in the computer's hardware and software places a further restriction on the variety of colors. Popular configurations devote 4 to 32 bit to the representation of color information, yielding 16 to 16,777,216 colors with different chromaticity and luminance. For a device-independent representation, these values need to be interpreted by a transformation into the CIE-XYZ system. This transformation assumes definite primary colors and scales the RGB values using a power-law to increase resolution in regions of the color space where the human eye is highly sensitive. Different transformations define different RGB color spaces. The standard RGB color space is sRGB. One can generally assume, in the absence of embedded profiles and any other information, that any 8-bit-per-channel image can be considered to be in the sRGB color space. However, every display has its own unique color signature, displaying a certain color according to manufacturing tolerances and material deterioration through use and age. Thus, displays offer an opportunity to adjust the mapping of the RGB values to the luminance of the RGB subpixels. Using a spectrophotometer, a correction matrix can be determined in a way the RGB values are displayed within a target RGB color space. RGB systems also include a reference white, considering that the human color perception system is able to adapt to varying illumination conditions. For instance, a white sheet of paper looks white in sunlight and in artificial light, but the XYZ values are different. A different light source causes a shift in the whole color space, to which the visual system is able to adapt. The CIE Standard Illuminant D65 is the reference white for the sRGB color space. The CIE 1931 color space chromaticity coordinates of D65 are $x = 0.31271$, $y = 0.32902$.

Although the CIE chromaticity diagram is a widely accepted standard to represent the color space, additional models are also popular in the context of human–computer interaction. One of them specifies hue, saturation, and brightness (sometimes the terms "luminance" or "lightness" are used instead of "brightness"). This model uses an upside-down cone to represent the color space (see Figure 8.6). On the edge of the cone's base, the visible light spectrum is arranged in a circle by joining red and violet. Hue is the actual color; it can be specified in percent, in angular degrees around the cone starting and ending at red (0° or 360°), or in 8-bit values (0–255). Saturation is the purity of the color, measured in percent or 8-bit values from the center of the cone (min.) to the surface (max.). At 0% saturation, hue is meaningless. Brightness is measured in percent or 8-bit values from black (min.) to white (max.). At 0% brightness, both hue and saturation are meaningless.

The CMYK model defines cyan, magenta, yellow, and black as main colors. This model is used to specify colors on the monitor for printing. In printing, colors are mixed subtractively and, using RGB, it would not be possible to produce many colors. By choosing cyan, magenta, and yellow as basic colors, many other colors, including RGB, can be produced. Theoretically, when all three basic colors are printed over each other, the resulting color should be black. In practice, however, this is not the case, and a fourth printing

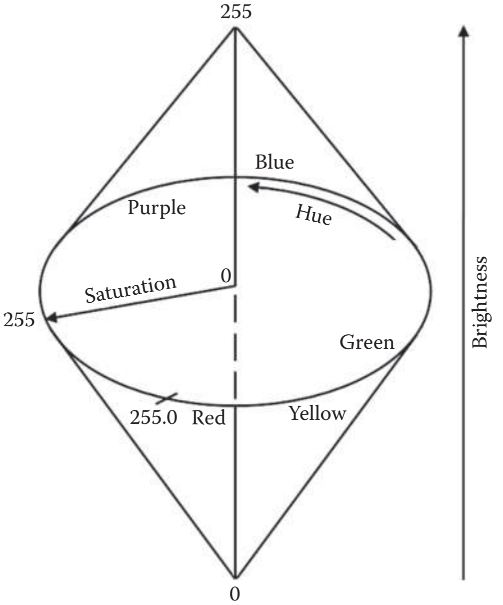


FIGURE 8.6 Hue, saturation, and brightness system for specifying color.

process with black ink must also be used. Values for CMYK are often specified as percentages.

8.3.2 Brightness

The brightness of a visual display depends not only on the optical power generated, transmitted, or reflected by the display but also on the response of the human eye at certain wavelengths. An adequate photometric measure is the photopic luminosity $K(\lambda) = 683.002$ lm/$W^*V(\lambda)$, which relates optical power in watts at a given wavelength to its effect on the human visual system. This luminosity can be considered as the optical spectral response of the eye of a "standard observer" (Nelson and Wullert 1997). Luminous flux Φ can be computed on the basis of $K(\lambda)$ and the spectral distribution of optical power $P(\lambda)$ by integrating the spectral response over the range of visible wavelengths (Macadam 1982):

$$\Phi = \int_{\lambda=380}^{\lambda=770} K(\lambda)P(\lambda)\mathrm{d}\lambda \tag{8.1}$$

The SI unit of luminous flux is the lumen (lm). The emitted luminous flux per unit solid angle Ω is called luminous intensity (I) (or radiant intensity) and is defined by

$$I = \frac{\mathrm{d}\Phi}{\mathrm{d}\Omega} \tag{8.2}$$

Luminous intensity is measured in candela (cd); the unit is lumen per steradian.

The human eye cannot collect all of the light that is radiated or reflected from a source. Brightness also depends on the size of the surface that the light is emanating from. Figure 8.7 shows how the brightness of a surface is actually given by the luminous flux per unit of the projected area of the emitting surface per unit solid angle depending on the viewing angle.

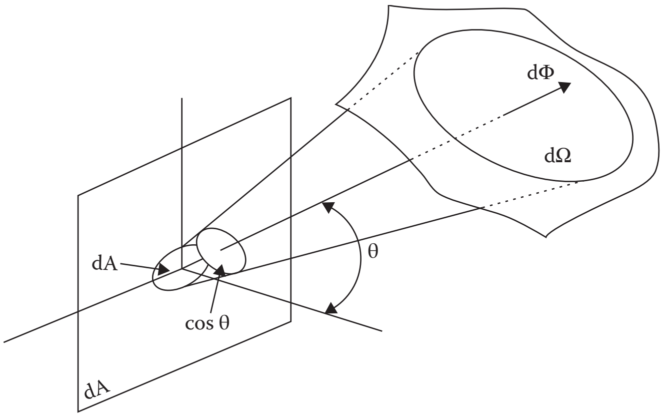


FIGURE 8.7 Definition of the brightness of a surface as a function of the direction from which the surface is observed. (Adapted from Nelson, T. J., and J. R. Wullert. 1997. *Electronic Information Display Technologies*. River Edge, NJ: World Scientific.)

Some displays such as LCDs appear dimmer from an oblique angle than from the normal viewing angle, whereas most emissive displays, such as CRTs, emit light in such a way that the angular luminous intensity approximately follows Lambert's cosine law (Lambertian surface), resulting in approximately constant luminance across all viewing angles.

Brightness of a visual display can be defined as the perceived amount of light traveling in a certain direction and specified by the photometric measure called luminance (L). Luminance is the luminous intensity of light (I) emitted from a light source per unit surface area (A) normal to the direction of the light flux (Çakir, Hart, and Stewart 1979). In this context, θ is the angle between the perpendicular from the surface and the direction of measurement (Boff and Lincoln 1988). It holds

$$L = \frac{\mathrm{d}I}{\mathrm{d}A\cos\theta} \tag{8.3}$$

The unit of luminance is lumen per steradian per square meter or candela per square meter (cd/m^2). In some cases, the unit foot-lambert (fL) is used to describe luminance (1 fL = 3426 cd/m^2).

8.3.3 Contrast and Gray Scales

Strictly speaking, contrast is not a physiologic unit, but it is nevertheless one of the most important photometry quantities. It is related to numerous visual performance criteria such as visual acuity or contrast sensitivity. There are different definitions of contrast.

For a luminance increment or decrement relative to background luminance (such as a single point), the contrast ratio C_r is used (Boff and Lincoln 1988):

$$C_r = \frac{L_{\text{object}}}{L_{\text{background}}} \tag{8.4}$$

The contrast ratio represents the factor by which one pixel is brighter than another pixel. For electronic information displays, maximum contrast ratio is used to determine the range of gray levels that can be displayed. Modern LCDs have a contrast ratio of 500:1 or more. Because LCDs do not emit light, the luminance in Equation 8.4 for contrast refers to the luminance of light either passing through the display (for a backlit transmissive type) or the luminance of the light that is reflected off the display's surface (for a reflective LCD). In multiplexed LCDs, the contrast is affected by the viewing angle. Therefore, the contrast should be indicated by referring to the solid angle, known as the "viewing cone" (Castellano 1992).

Large contrast ratios are also needed to satisfy current gray scale requirements. Following from the idea that the brightest areas are white and the darkest areas are black, levels of brightness in between the two extremes are referred to as gray levels or gray shades, and the ability to display them as gray scale (Nelson and Wullert 1997). Technically, gray scale is a term that should be applied only to monochrome or "gray" displays. However, the term is often applied to color displays where intermediate brightness controls are required by the system (Castellano 1992). The number of gray scales is determined both by the contrast level and the ability of the human visual system to distinguish between the different brightness levels. Our visual system reacts to changes in brightness level as a logarithmic function, so it may not perceive very small differences in brightness (Theis 1999). The acceptable difference in brightness levels between scales is 1.414 (the square root of 2). In order to obtain five levels of gray scale above background, a display must have a contrast ratio of at least 5.6:1.0 (1.00, 1.41, 2.00, 2.82, 4.00, and 5.65:1.00) (Castellano 1992). Full-color displays have about 128 to 256 linear gray levels. The number of gray shades (G) that can be displayed can be defined as a logarithmic function based on contrast ratio (Nelson and Wullert 1997):

$$G = 1 + \frac{1}{\log\sqrt{2}} \log\left(\frac{L_{\max}}{L_{\min}}\right) \tag{8.5}$$

$L_{\max}$ and $L_{\min}$ are maximum and minimum luminance. TFT LCDs improve contrast ratio by dimming the backlight for dark images. The contrast ratio of a black pixel displayed with dimmed backlight and a white pixel displayed at full backlight is called the dynamic contrast.

Different contrasts measures are used to quantify overall contrast within an image. In cases where small features are present on a large uniform background, the Weber Contrast C_w (luminance contrast) is used. The International Lighting Commission defines as follows:

$$C_w = \frac{L_{\text{object}} - L_{\text{background}}}{L_{\text{background}}} \tag{8.6}$$

The *Michelson Contrast* C_m (also called modulation contrast, depth of modulation, or relative contrast about the mean luminance) is generally used for periodic stimuli that deviate symmetrically above and below a mean luminance value, such as gratings or bar patterns. It is computed as follows (Boff and Lincoln 1988):

$$C_m = \frac{L_{\max} - L_{\min}}{L_{\max} + L_{\min}} \tag{8.7}$$

The Michelson Contrast will take on a value between 0 and 1. The minimum modulation threshold, meaning the smallest detectable brightness modulation, occurs at a brightness modulation of approximately 0.003 at about three cycles per degree of the visual field (Nelson and Wullert 1997).

Sanders and McCormick (1993) specify another possibility for defining contrast (called luminous contrast) as the difference between maximum and minimum luminance in relation to maximum luminance:

$$C_l = \frac{L_{\max} - L_{\min}}{L_{\max}} \tag{8.8}$$

These contrast measures can be converted from one to another. For example, given the contrast ratio C_r and knowledge of positive contrast conditions ($L_{\max} = L_{\text{object}}$; $L_{\min} = L_{\text{background}}$), the other visual contrasts can be calculated as follows:

Weber Contrast:

$$C_w = C_r - 1 \tag{8.9}$$

Michelson Contrast:

$$C_m = \frac{C_r - 1}{C_r + 1} \tag{8.10}$$

Luminous Contrast:

$$C_l = \frac{C_r - 1}{C_r} \tag{8.11}$$

8.3.4 Glare and Light Reflection

High-luminance levels in the field of view cause glare discomfort. Glare caused by light sources in the field of view is called direct glare; glare caused by light being reflected by a surface in the field of view is called reflected glare (Figure 8.8).

Reflected glare can occur from specular (smooth, polished, or mirror-like) surfaces, spread (brushed, etched, or pebbled) surfaces, diffuse (flat or matte) surfaces, or as a combination of the above three (compound) (Sanders and McCormick 1993). Glare sources are more disturbing when they have higher luminance and when they are closer to the fixation point (Sheedy 2005). Experiments also show that visibility is decreased by glare, and the decrease is greatest when the source of the glare is in the direct line-of-sight (Boff and Lincoln 1988).

To avoid glare, it is advisable to position the display at a right angle to the window (so that the line of vision is parallel to the window). The display can be protected with curtains, blinds, or movable walls. Lamps that can be reflected in the monitor must not have a mean luminance of more than 200 cd/m^2, and the maximum luminance must be less than 400 cd/m^2 according to German standard DIN 5035-7. An example of good workplace design is shown in Figure 8.9.

8.3.5 Resolution

The arrangement of pixels into rows and columns is defined as the pixel format. It is often referred to as a resolution; however, this is not a resolution parameter by itself. For instance, the resolution of a flat-panel display mainly depends on its screen diagonal and its dot pitch (stripe pitch, SP). When set to lower resolutions, a pixel encompasses multiple dots. Thus, the resolution in terms of pixel density, or the number of pixels per linear distance (pixels per inch or pixels per centimeter). This parameter indicates how close the pixels are.

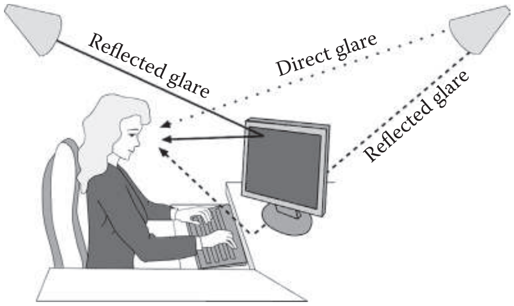


FIGURE 8.8 Direct and reflected glare.

As a picture element, a pixel is the basic unit of programmable color on a display. In classic color displays, the pixel is usually formed from a number of dots and it may consist of a number of triads, which are composed of red, green, and blue dots. The dot pitch in a CRT display with a shadow mask is defined as the distance between the holes in the shadow mask, measured in millimeters (mm). In matrix-driven, flat-panel displays every single pixel is composed of a red, a green, and a blue phosphor dot or filter element, and dot pitch is defined as the center-to-center distance between adjacent green phosphor or filter element dots. Thus, the pixel density is the reciprocal of pixel pitch, which is equal to the dot pitch. Resolution quality levels are defined as follows (Castellano 1992):

- Low resolution: <50 pixels per inch
- Medium resolution: 51–70 pixels per inch
- High resolution: 71–120 pixels per inch
- Ultrahigh resolution: >120 pixels per inch

Obviously, ergonomic resolution requirements depend on the viewing distance to the display. The visual acuity of a person with normal vision and under normal ambient light intensity is frequently considered to be what was defined by Snellen as the ability to recognize a standardized symbol (optotype) when it is subtended 5 minutes of arc. Therefore, if the distance from the display to the eye of the observer is 50 cm, he or she will be able to distinguish approximately 175 pixels per inch. The quality levels mentioned above refer to displays on a desktop. For handheld displays, viewing distances are typically closer, up to 20–25 cm. Therefore, the resolution of current handheld displays often far exceeds the level for ultrahigh resolution. For HMDs specifying the

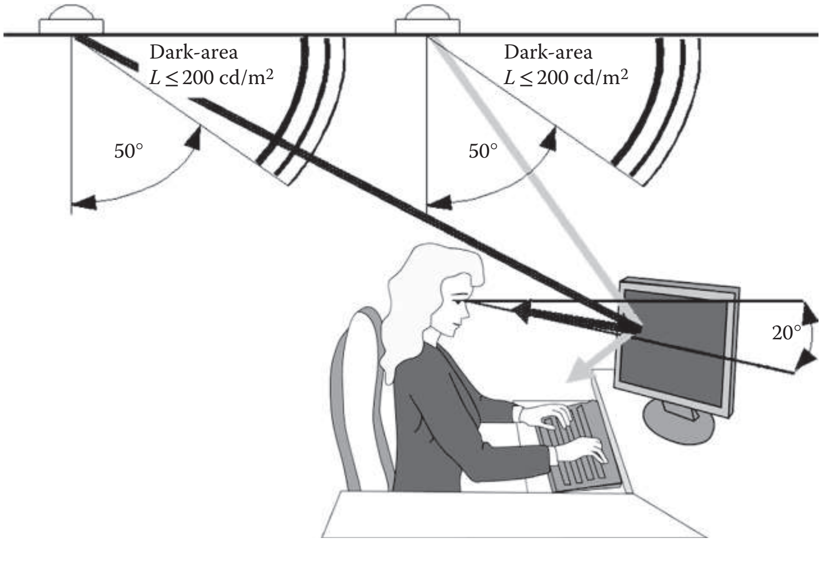


FIGURE 8.9 Glare can be avoided by limiting lamp luminance and by positioning the display correctly.

resolution in pixels per inch is often not meaningful because optics magnify the physical screen to appear at a predefined distance from the observer and to cover a specific field of view. For this reason, HMD resolution is usually specified as the number of pixels per degree of arc. HMDs typically offer 10–20 pixels/°, though advances in microdisplays will significantly increase this ratio in the near future. A higher resolution can also be reached at the cost of the width or height of the field of view. Human visual acuity is reached at approximately 60 pixels/°.

8.3.6 Refresh Rate and Response Times

The number of times that the image on the display is drawn per second is called the display refresh rate (also called vertical scanning rate on CRTs). The unit of measurement is hertz (Hz). On CRTs, a high-display refresh rate prevents a flickering image because there is only a small amount of time between two successive stimulations of a single dot, where luminance fades away after several milliseconds. Because the retina retains a precise image of an object for about 10–20 milliseconds after an object has disappeared, this temporary disappearance of the dot will not be noticed if the dot is stimulated at a fast enough rate. On CRTs, a refresh rate of between 70 and 80 Hz is needed to ensure a flicker-free image on the display (meaning the flicker is not perceivable) (Bosman 1989).

The effects of the "refresh rate" on human vision are different for other display types in which the color and luminance brightness of the pixels is constant until the state of the pixels is explicitly updated. One example is a classic LCD, whose shutters remain at the opacity level last addressed. Another example is an electrophoretic display, whose charged pigment particles stay in a stable configuration to form a visible image. For these kinds of displays, the response time needed by the pixels to transition to a new visible state is a more appropriate quality metric. Long response times will cause motion blur when displaying scenes with a high-dynamic range. But motion blur is also an issue if response times are short and the frame rate at which the dynamic scene is rendered is low. There is a simple reason for this effect: If an object has moved in two sequential frames and it disappears at one position on the display to appear at another position, the retina keeps the imprint of the object at the old position for 10 milliseconds, and the user perceives both pictures simultaneously. At a low frame rate of about 30 frames per second, the visual system will perceive the images of two frames with an interval of about 33 milliseconds. These pictures differ substantially for dynamic scenes and therefore blur perception. At higher frame rates, the two images differ less and the impression of motion blur is reduced. Interestingly, motion blur is not that critical on CRTs if the refresh rate is set to 60–80 Hz. This is because the pixels on CRTs turn black before they are re-lit, so the imprint of the old picture is reduced. Manufacturers of LCDs attempt to simulate CRTs by using a strobing backlight. If the refresh rate is high, CRTs also tend to show motion blur at low frame rates. Some displays that run at 100 Hz or more add additional functions to avoid motion blur. Motion interpolation can cut the amount of blur by inserting extra synthesized in between frames.

8.3.7 Depth in 3D Displays

Depth perception depends on physiological sensations that are addressed differently by different display technologies. Monocular depth cues like perspective, occlusion, motion parallax, and comparison of familiar size can also be displayed on standard displays and refer to the structure of the image. Binocular depth cues provide depth information when viewing a scene with both eyes. Because they cannot be addressed by standard displays, new technologies need to be considered. Both eyes see a real scene from two different observation points, which are fused by the brain into a single visual impression that includes depth information. The two additional types of depth information achieved with binocular vision are convergence and retinal disparity. Convergence is a binocular oculomotor cue referring to the orientation of the eyes. If visual attention is focused upon an object, both eyes will adjust their orientation to perceive the image of the object on the fovea of the retina. The gaze direction of both eyes crosses at the depth of the object's location. Retinal disparity refers to the effect that if both eyes focus on the same object, the images they see are from a slight different perspective. The visual system determines corresponding points of the two images by pattern matching. Two pairs of corresponding points on the retina relate to two different visual features of objects in the physical space. The different offsets of the two pairs of corresponding points can be utilized by the visual system to determine the relative depth of two visual features. In general, display technology must be capable of showing two independent pictures to each eye of a user to produce binocular depth cues. This can be achieved by using common displays with optics that inverse multiplex two images for the left and the right eye (see Section 8.4.10). Depending on the technology, inverse multiplexing may not be perfect, in which case each eye will tend to catch a glimpse of the picture addressed to the other eye. This effect is called ghosting, and it deters visual impression. The acuity of binocular depth cues is closely related to the visual acuity of the single images for each eye. If the resolution of each single image exceeds the limits of human vision, the acuity of disparity cannot be further improved. However, there are individual differences in how well binocular depth cues could be resolved. A small group of people (about 2%) are not able to see 3D using stereo vision. One might expect that if both eyes were presented with two pictures that are arranged in such a way that each eye sees the picture it would have seen in a real scene, depth perception would also be close to reality. But one must consider that in reality the eyes also need to change the optical power of their lenses to perceive a clear image of objects at different distances. As long as both images are displayed on a display surface at a distinct distance, accommodation stays constant while convergence changes to focus on objects with a different parallax (different depth) in the two displayed images. This mismatch causes eyestrain and limits the range of parallax that can be used in a pair of stereo images to encode depth (Lambooij, IJsselsteijn, and Heynderickx 2007). Therefore, when using simple stereoscopic displays, objects cannot be displayed to appear at arbitrary distances to the given display screen. Furthermore, this mismatch causes distorted depth perception (Armbrüster et al. 2008; Kleiber and Winkelholz 2008; Patterson, Moe, and Hewitt 1992). To overcome these limitations, other technologies are needed that exactly reproduce the lightfield of real 3D objects. Holographic and volumetric displays are two promising approaches (see Section 8.4.10). Volumetric displays are able to address single points as voxels in a 3D display space and make them emit or transmit light as small diffusive point sources. The resolution of a volumetric display is then given by the number of voxels that can be addressed and the display volume. Volumetric displays can also cope with motion parallax if the user moves his or her head. Virtual objects can be seen from different viewpoints just by moving the head as if the virtual objects were real. However, human performance in perceiving and processing depth information from a visual field is also limited in reality (Winkelholz, Kleiber, and Schlick 2010a,b). Acuity in depth perception is far lower than visual acuity in lateral dimensions (Norman et al. 1996).

8.4 TECHNOLOGIES

8.4.1 Taxonomy of Electronic Information Display Technologies

Figure 8.10 provides a systematic overview of display technologies (Theis 1999). All of these technologies can be assigned to one of the three basic display concepts: direct-view, projection, and offscreen.

8.4.1.1 Direct-View Displays

These are electronic information displays in which the image produced by a display device is viewed directly without first bouncing off a screen. All plasma televisions and TFT LCD computer monitors are direct-view displays. These displays tend to work well in bright light conditions and have greater luminance than projection displays.

8.4.1.2 Projection Displays

Unlike direct-view systems, a projection display relies on the projection of an image onto a screen. There are front and rear projection systems, which mainly differ with regard to screen technology. Front projection utilizes a reflective screen surface, whereas rear projection uses a transmissive screen material, typically transparent plastic sheets. Projection displays work best in dimly lit environments. In particular, a front projection setup requires a darkened room for optimal viewing quality.

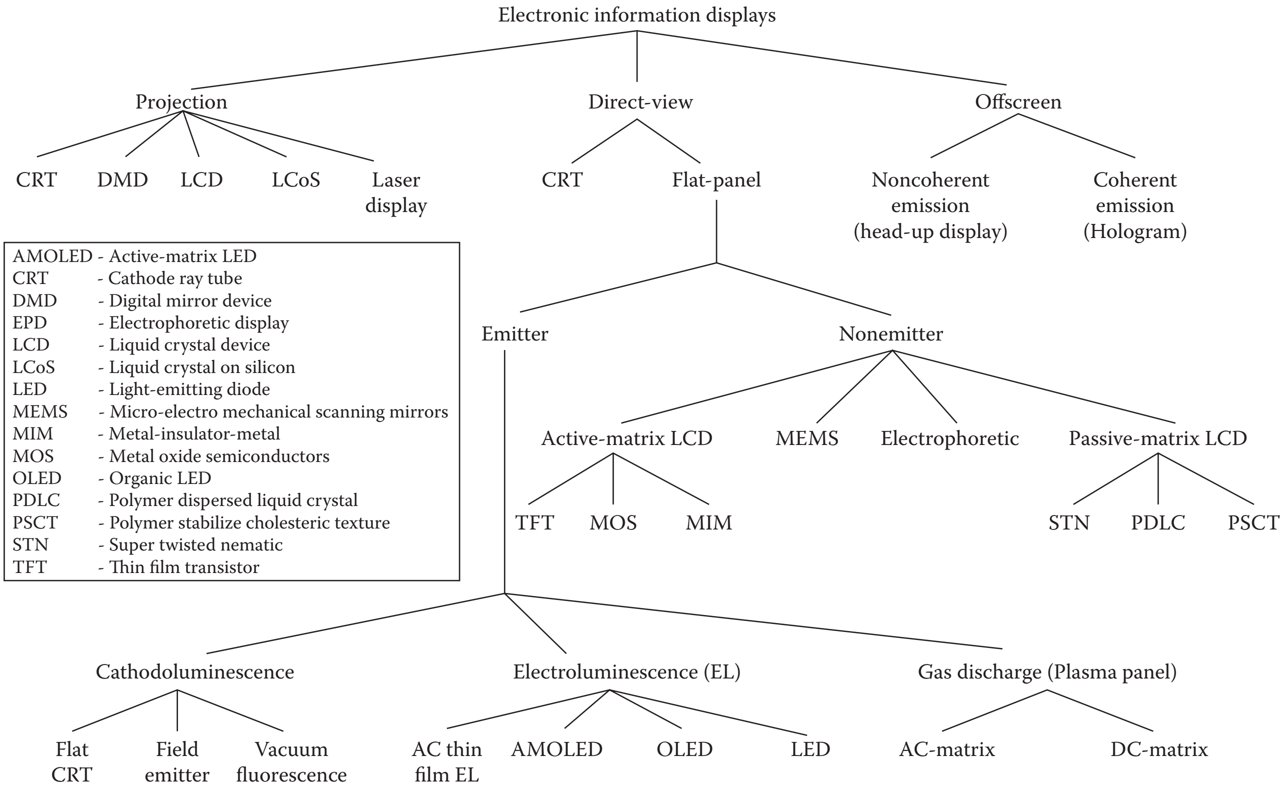

Electronic information displays
Projection
Direct-view
Offscreen
CRT
DMD
LCD
LCoS
Laser display
CRT
Flat-panel
Noncoherent emission (head-up display)
Coherent emission (Hologram)
AMOLED - Active-matrix LED
CRT - Cathode ray tube
DMD - Digital mirror device
EPD - Electrophoretic display
LCD - Liquid crystal device
LCoS - Liquid crystal on silicon
LED - Light-emitting diode
MEMS - Micro-electro mechanical scanning mirrors
MIM - Metal-insulator-metal
MOS - Metal oxide semiconductors
OLED - Organic LED
PDLC - Polymer dispersed liquid crystal
PSCT - Polymer stabilize cholesteric texture
STN - Super twisted nematic
TFT - Thin film transistor
Emitter
Nonemitter
Active-matrix LCD
MEMS
Electrophoretic
Passive-matrix LCD
TFT
MOS
MIM
STN
PDLC
PSCT
Cathodoluminescence
Electroluminescence (EL)
Gas discharge (Plasma panel)
Flat CRT
Field emitter
Vacuum fluorescence
AC thin film EL
AMOLED
OLED
LED
AC-matrix
DC-matrix

FIGURE 8.10 Classification of electronic information technologies with high information content. (Adapted from Theis, D. 1999. Display technologie. In *Vom Arbeitsplatzrechner zum ubiquitären Computer [From the Desktop PC to the ubiquitous Computer]*, ed. C. Müller-Schloer and B. Schallenberger, 205–38. Berlin: VDE.)

8.4.1.3 Offscreen Display Systems

These display systems do not use a projection screen. Instead, a natural medium like windshield made of laminated safety glass or even the retina can be used for image projection. Offscreen systems are based either on coherent or noncoherent light emission. Coherence is a property of waves that measures the ability of the waves to interfere with each other. Laser light usually has far greater coherence than nonlaser light. VRDs (see Section 8.4.7) and 3D holographic head-up displays are examples of offscreen display systems.

8.4.2 Cathode Ray Tube

CRTs are the classic cathodoluminescent display. Light is generated by exciting a luminescent material with energetic electrons. A CRT consists of a glass bulb, a cathode, an electron gun, a deflection yoke, a mask, and a phosphor coating. An electron gun located in the back panel of the device emits negatively charged electrons that are attracted and accelerated by an anode located in front of the screen. The electron beam is diverted by an electromagnetic field built up by the deflection coils and thus directed toward the screen. Electrons are extracted from the cathode by thermal emission from low-surface-potential materials (typically metallic oxides), and the electron beam generated at the cathode is then accelerated, deflected, and focused by a series of electrostatic lenses and deflection coils. A screen mask is attached in front of the ground-glass plate so that the electron beam is focused and then steered onto the phosphorus layer deposited on the front surface. Display screen masks are either dot-mask screens, Trinitrons, or slot mask screens.

CRT technology was used in both color and monochrome systems because it offered high-information content at low costs and wide viewing angles from off-axis. Today flat-panel systems have replaced this display technology in most application areas because color CRT systems are usually very large and heavy, have high-power consumption and operating voltage, low vibration robustness, and a maximum screen size of only 40 in. Domains where CRTs are still popular include the printing and broadcasting industries, which prefer them for displaying photos, videos, and graphics because of their high pixels per unit area, color fidelity, contrast, and correct color balance (Castellano 1992; Ozawa 2007).

8.4.3 Liquid Crystal Displays

LCDs have become increasingly popular in recent years and currently hold the largest share in the market. Early commercial developments of LCDs concentrated on small numeric and alpha-numeric displays that rapidly replaced LEDs and other technologies in applications such as digital watches and calculators (Bosman 1989). Now there are ultrahigh-resolution displays for use in many applications such as laptops, handheld computers, HMDs, and miniature televisions (like those in airplane seats). LCDs have the following advantages:

- Their power consumption is low.
- They operate at low voltages.
- Their lifetime is very long in normal environments.
- Displays may either be viewed directly in transmission or reflection, or may be projected onto large screens.

LCDs consist of two glass plates with microscopic lines or grooves on their inner surfaces and a liquid crystal layer between them. Liquid crystal materials do not emit light, so external or back illumination must be provided. This circumstance theoretically puts LCDs among other nonemitter display technologies (e.g., electrophoretic displays, see Section 8.4.8), but the tight integration with an extra light emitter allows LCD technology to firmly stand on its own. The physical principle is based on the anisotropic material qualities of liquid crystals. When substances are in an odd state that is somewhat like a liquid and somewhat like a solid, their molecules tend to point in the same direction like the molecules in a solid, but they can also move around to different positions like the molecules in a liquid. This means that liquid crystals are neither a solid nor a liquid but are closer to a liquid state. LCDs operate by electrically modulating the anisotropy between optical states in order to produce visible contrast. In an electric field, liquid crystals change their alignment and therefore their translucence. If no voltage is applied, light can pass through and the pixels appear bright. When voltage is applied, the pixels become dark. The light to be modulated may either originate from ambient light or from additional bright light sources placed behind the LCD or at the sides (Lueder 2010).

The two principal flat panel technologies are the passive-matrix LCD and active-matrix (AM) LCD. Passive matrix addressing is used in twisted nematic (TN) LCDs. In TN LCD displays, the microscopic lines of the glass plates are arranged orthogonally to each other, and the glass plates serve as polarizers (Precht, Meier, and Kleinlein 1997). Their directions of translucence lie at right angles on top of one another so no light can pass through (see Figure 8.11). Because of the fine grooves on the inner surface of the two glass panels (arranged vertically on one panel and horizontally on the other), the liquid crystal is held between them and can be encouraged to form neat spiral chains. These chains can alter the polarity of light. In the so-called nematic phase, the major axes of the crystal's molecules tend to be parallel to each other.

Nonpolarized light from background illumination can pass the polarization filter with just one plane of polarization. It is twisted about 90° along the helix and can thus pass through the second polarization layer. When there is no electric current, the display appears to be bright. Applying an electric current to TNs causes them to untwist and straighten, changing the angle of the light passing through them so that it no longer matches the angle of the top polarizing filter. Consequently, no light passes through that portion of the LCD; it becomes darker than the surrounding areas, and the pixel appears black.

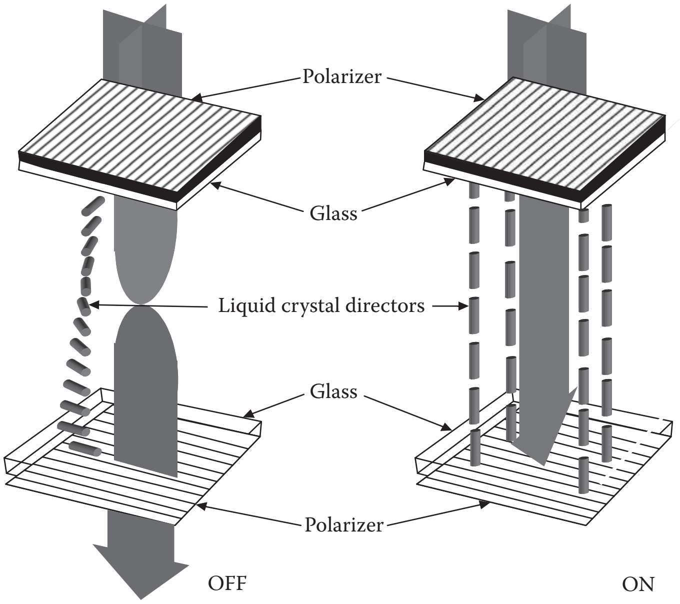


FIGURE 8.11 Principle of operation of a twisted nematic LCD.

Applying different electric currents allows gray scales to be produced with LCD technology. One disadvantage of this is the relatively low contrast, but that can be improved by applying a steep electro-optic characteristic line of the liquid crystals. Low voltage is then sufficient to change the translucence, causing the liquid crystals to twist by more then 90º. Such displays are called super twisted nematic, double super twisted nematic, or triple super twisted nematic (Schadt 1996).

In passive-matrix LCDs, the electric field expands over the pixel to be addressed to the entire horizontal and vertical electrodes. This results in annoying stripes or ghosting. Other disadvantages are slow response times and a prelightening of the pixels.

AM addressing places an electronic switch at each pixel of an LCD thus controlling the charging of the pixel capacitor up to the voltage corresponding to the desired shade and then holding this voltage until the next image information is written in. The available switches are TFTs acting like diodes (Lüder 2003). To display a full range of colors, each element on an LCD is assigned to a primary color by a special filter deposited on the faceplate of the LCD. To guarantee high contrast, brightness, and high quality of color representation, many transistors are needed. For example, a resolution of 1920 × 1080 pixels requires a total of 6.22 million transistors—one per primary color in each subpixel.

The traditional representation errors caused by an incorrect deflection do not occur because of the fixed matrix, but defective transistors can cause errors, leading to permanently bright or dark pixels. These pixels are distracting, particularly when all three transistors of the subpixels are switched to bright, which causes a white pixel to appear on the screen.

CRTs easily represent black; the cathode simply does not emit electrons. LCDs, however, have to completely block out the light from the backlight. Technically, this is impossible, and as a consequence the contrast is reduced. Furthermore, it is often said that CRTs are more suitable for displaying scenes with fast-moving objects. This is true if the response latency of the LCD is slow or the scene is displayed at a low frame rate. Most LCD displays available today have overcome these limitations (see Section 8.3.6).

When an LCD display is viewed from an angle, it appears darker and color representation is distorted (see Section 8.3.2). It is worth mentioning here that the restricted viewing angle has a positive effect on privacy, as in the case of automated teller machines (ATMs). Technologies such as in-plane switching, multidomain vertical alignment, and TFTs have improved the size of the viewing angle. Typically only the color becomes less saturated when viewing at extreme angles. Although many technical improvements to viewing angles have been done on lateral angles, vertical angles still often lead to non-negligible distortion. The nominal screen diagonal of an LCD is equivalent to the effective screen diagonal. In contrast, CRT nominal screen diagonals are smaller than the effective ones.

The well-known liquid crystal on silicon (LCoS) displays are also nonemitter displays. The advantages of LCoS technology are high luminance, high resolution, high contrast, relatively low production costs, and economy of weight. Unlike LCD technology, LCoS displays do not use light transmission through glass layers to create an image on a screen. Instead, an active array of pixels with liquid crystals is directly mounted on silicon that has been coated with aluminum. This additional layer consists of a reflective passivation layer that reflects the incoming RGB light toward a prism. The prism directs the light at a projection target, such as a flat screen or projection field. The crystals' orientation relative to the reflective surface can be controlled using an electric current. The current either brings the crystals into a reflective state or aligns them so that no light is reflected.

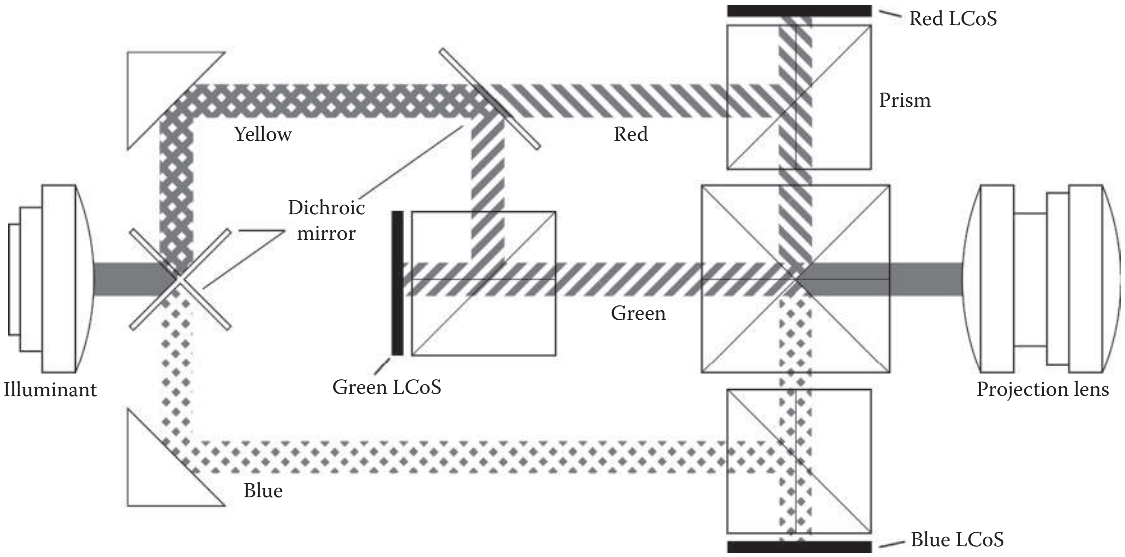


FIGURE 8.12 Conceptual diagram of three-panel liquid crystal on silicon display. The diagram shows the path and the split of the light beam into two beams of different wavelengths with the help of dichroic mirrors.

The light beam from the illuminant is often separated into blue and yellow components by dichroic mirrors (see Figure 8.12). The blue light beam is directed through a prism onto a blue LCoS chip, while the yellow beam passes through another dichroic mirror that splits it into its red and green components. These distinct beams are directed through additional prisms onto additional chips for red and green colors. A prism located directly in front of the LCoS panels directs the light beam toward the combining prism. The combination of the three reflected beams is then passed through a projection lens that transmits a full-color image onto the desired screen.

Two different LCoS display designs are popular—three panel design and one panel design. Three panel design with distinct RGB LCoS chips was explained above. This design distributes the three primary colors among different panels, so all three panels need to be aligned very accurately to precisely create the desired image dots. One panel design uses only one LCoS chip showing the RGB image components by temporal multiplexing. In this case, the observer's eyes put together the color stream. During image generation, a color wheel lightens the display with either red, green or blue light. If the color field frequency drops below about 540 Hz, a so-called color breakup can occur. A color breakup causes viewers to perceive false colors for a short period of time. Color breakup can also occur when using single-chip projectors.

The three-panel design is frequently used in digital light processing projectors for its high luminance and very good contrast. LCoS is not a very common technology for visual displays at present. However, LCoS offers the possibility of building very large high-definition screens for public displays and home entertainment. In digital light processing projectors, an image can also be created using microscopically small mirrors laid out in a matrix on a semiconductor chip. This technology is known as a DMD (see Figure 8.10). Each mirror generates one or more pixels in the projected image. The number of mirrors determines the image resolution. The small mirrors can be positioned rapidly to reflect the incoming ray of light through a lens (Hornbeck 2001).

8.4.4 Plasma Displays

An electrical discharge in gas is the oldest electro-optical phenomenon capable of producing light (Bosman 1989). Millions of years elapsed before this effect was identified, analyzed, and mastered by humans. The first attempts to produce a matrix display panel were made in 1954. Since then, research has continued and a host of approaches have evolved (Bosman 1989). The advantage of this technology is that, unlike front view projection screens, the room lights do not have to be off for people to see the image clearly and easily. This means plasmas are excellent for video conferencing and other presentation needs (Pioneer 2001).

With plasma technology, two glass plates are placed with their parallel thin conducting paths at right angles to one another (Precht, Meier, and Kleinlein 1997). The gap between the plates is evacuated and filled with a gas mixture (see Figure 8.13). If sufficiently high voltage is applied to the cross point of two orthogonal conducting paths, the gas ionizes and begins to shine (like many small gas discharge lamps). The inside of one glass plate is coated with a phosphorus layer, which is, according to the fundamental colors, composed of three different kinds of phosphorus. There is a wired matrix below the phosphorus-layer to trigger the PDP. The space between the glass plates is divided into gas-filled chambers. When voltage is applied to the wired matrix on the bottom of the display, the gas is transformed into a plasmatic state and emits ultraviolet radiation, causing the phosphorus to glow.

The advantage of this technology is that it can produce large, flat screens that perform extraordinarily well under most ambient light conditions. Even very bright light does not wash out the image on these screens. Another characteristic

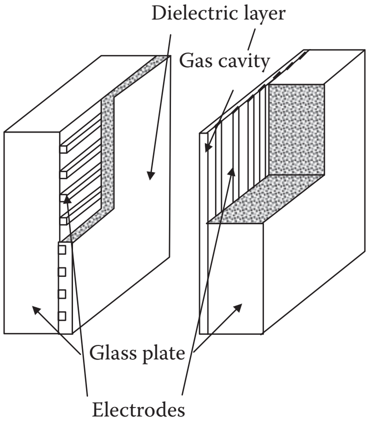


FIGURE 8.13 Major parts and components of a plasma display panel.

of a plasma panel is its extreme viewing angles, both vertically and horizontally. With a 160° viewing angle, people sitting to the side of the screen are still able to see the image without losing any information (Pioneer 2001).

Plasma screens also have disadvantages. They consume a large quantity of power, making the technology unsuitable for battery-operated and portable devices. Further, high voltage is required to ignite the plasma, and the phosphorus layers degrade over time. Finally, the pixels are oversized, so users must be some distance from the display.

8.4.5 Electroluminescent Displays

Electroluminescence (EL) refers to the generation of light by nonthermal means in response to an applied electric field that produces light (Nelson and Wullert 1997). In early ELDs, the phosphors were used in powder form like in CRTs. Current ELDs use a thin film of phosphorescent substance (yellow-emitting ZnS: Mn) sandwiched between two conducting electrodes. The top electrode is transparent. One of the electrodes is coated with vertical wires and the other with horizontal wires, forming a grid. When electrical current passes though a horizontal and a vertical wire, electrons are accelerated and pass through the phosphorous layer. Light is emitted as a pulse by the excited activator atoms. The activators are transition metal or rare-earth atoms. Once excited, these centers can decay to the ground state through a radiating transition, and thus emit light with their specific emission spectrum (Budin 2003). EL can also be obtained from organics. Organic EL devices generally consist of a hole transport layer and an electron transport layer between electrodes. The radiating recombination can be further optimized by introducing fluorescent centers near the interface of the two layers. Such structures are particularly useful in tuning EL colors (Theis 1999).

The main concern for ELDs is high-quality color presentation. A large number of laboratories have experimented with many materials, activators, and full-color solutions (Ono 1993). Color ELDs can be manufactured in different ways. One of these approaches, additive color synthesis, is quite common for emitters and involves using three juxtaposed patterned phosphors. However, this approach suffers from a reduced spatial fill factor for each monochromatic emitter. The second approach to full-color ELDs consists of using a single white-emitting structure with patterned color filters. It requires a much simpler manufacturing process. A hybrid solution, which is being developed into a full-color commercial product at Planar, involves stacking and registering several plates (King 1996). The first plate is a glass plate with the active structure on its far side, and transparent electrodes on both sides of a patterned ZnS:Mn (filtered to red) and ZnS:Tb (green) structure similar to that described before. The second glass plate has a fully transparent blue-emitting Ce:Ca thiogallate structure on top of the plate. On both plates, row electrodes are reinforced by a thin metal bus (Theis 1999; Budin 2003).

There are many different types of EL technology and alternating current thin film electroluminescent displays are probably the most commonly used ELDs. These days displays of 1280 × 1024 pixels with more than 1000 lines per inch (lpi) can be manufactured cheaply and are used for advertising applications such as electroluminescent billboards and signs. ELD offer several advantages: sharp pixel edges, good contrast (10:1 at 400 lux), a wide viewing angle, fast response time, durability, shock resistance, operation at high and low temperatures, and the smallest thickness and lowest weight compared with other flat panel displays (Budin 2003). A significant disadvantage of ELDs is their limited ability to display full color and the relatively high voltage (60–600 V) needed for operation. For battery-operated devices, this voltage must be generated by a converter circuit within the display (Nelson and Wullert 1997). At first, ELDs were primarily used for expensive instruments, but current technical advances allow wide application in industry and service areas.

EL devices also include LEDs. LED technology is used in almost every consumer electronic product on the market. The operational principle of LEDs can be summed up as follows: with no voltage or reversed voltage applied across the pn junctions, an energy barrier prevents the flow of electrons and holes. When a forward bias voltage (1.5–2 V) is applied across the junction, the potential barrier height is reduced by allowing electrons to be injected into the p region and holes into the n region. The injected minority carriers recombine with carriers of the opposite sign, resulting in the emission of photons (Castellano 1992). LEDs as well as some types of ELDs can also be used as a backlight for LCD screens (see Section 8.4.3).

A novel application of the EL principle is OLEDs, whose emissive electroluminescent layer is composed of a thin film of organic semiconductor material. This layer is formed between two electrodes, and at least one of the electrodes is transparent. Unlike inorganic solid-state based LEDs, OLEDs typically emit less light per unit area. However, the absence of brittle materials and the direct emission of colored light enable the production of very thin, flexible displays (<1 mm). Their use in handheld devices is advancing quickly because

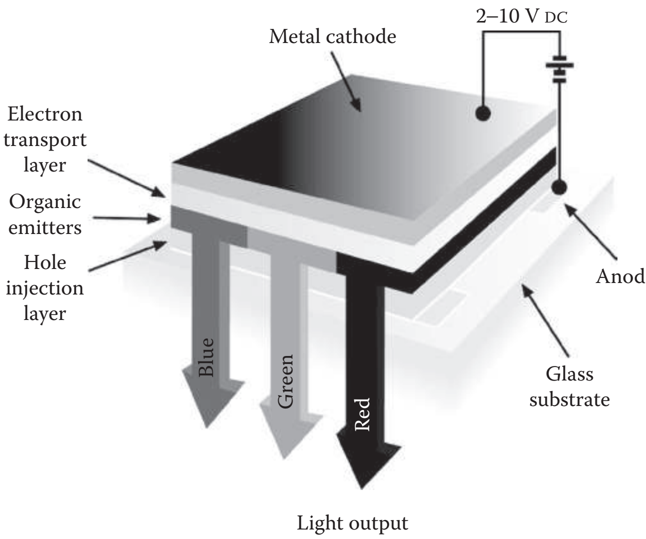


FIGURE 8.14 Layout and operating mode of an active-matrix organic light-emitting diode panel.

of their structural versatility and lower power consumption. OLEDs can be combined with a TFT backplane to actuate the display. Active-matrix OLEDs (AMOLED) are based on an array of thin-film transistor switches that control the state of each of the pixels. RGB subpixels can be either stacked on top of one another or next to another (see Figure 8.14). AMOLEDs provide a higher refresh rate than their passive-matrix OLED counterparts, and they consume significantly less power. AMOLEDs can also have a faster response time than standard TFT LCD screens (up to a 100,000 Hz refresh rate). The amount of power the display uses varies depending on the color and brightness shown (Yagi et al. 2008).

The biggest technical challenge for OLEDs is the limited lifetime of the organic materials. The layers are prone to material degradation and water can damage the organic compounds in the displays. Therefore, improved sealing processes are important for manufacturing. The differential color output has to be adjusted after some usage period, as the OLED material used to produce blue light degrades more rapidly than the materials that produce other colors, and the misaligned color balance becomes noticeable (Fellowes et al. 2009).

8.4.6 Cathodoluminescent Displays

The working principles of these displays are similar to those of a CRT. As a result, they are also referred to as flat CRTs. One of the most successful flat CRT configurations is a vacuum fluorescent display (VFD). It is essentially a triode with arrays of grids and phosphor-coated anodes used for coincident addressing. The grid structure is positioned over the anode and the cathode filaments are stretched and held above the grid. The entire structure is then sealed in an evacuated cell. The electrons, which are emitted by cathode filaments, are controlled by grids. When a grid is supplied with a positive voltage, it attracts the negative electrons; positive electrons are attracted when the grid is supplied with negative voltage. The electrons collide with the phosphor-coated anode, resulting in photon emission (Castellano 1992; Nelson and Wullert 1997).

One of the disadvantages of VFDs is that their manufacture cannot easily be scaled to large sizes because the front and back plates are only supported at the edges. Another problem is equalization of the luminous efficiencies. The blue phosphors tend to be very low in luminous efficiency, making energy efficient use of a full-color matrix VFD difficult. The technology remains promising in narrow application areas because it is possible to create a Lambertian light with 180° viewing cones, and it is not limited by the response of rearranging liquid crystals. It is therefore capable of normal function in subzero temperatures, making it ideal for outdoor devices in cold climates (Ozawa 2007).

8.4.7 Laser Displays

LASER is the acronym for light amplification by stimulated emission of radiation. Laser light is generated in the active medium of the laser. Energy is pumped into the active medium in an appropriate form and is partially transformed into radiation energy. In contrast to thermal emitters, a laser emits a concentrated and monochromatic light with high local and temporal coherence.

In contrast to conventional light sources, laser light has rays that are spatially coherent and narrow low-divergence beams that can be manipulated with lenses. The emitted light is focusable to a specific wavelength and has a high-power density, a monochromatic character (light of one wavelength), and high local and temporal uniformity (the same phase). These characteristics can be used to generate an image in the same way as a conventional CRT system: The laser is directed row by row across the display like conventional rear projection systems. The systems can be classified into two types: single scanning and dual scanning. In a single-scanning mirror system, one mirror directs the laser beam, drawing horizontal lines back and forth to create a frame of the moving image. At the end of each line, the beam shifts vertically. In a dual-scanning mirror system, one mirror directs the laser light back and forth along a horizontal path and then sends it to another mirror that adjusts the vertical position (Lincoln 2010). Called laser video displays, these devices are capable of displaying 90% of the color spectrum that the human eye can potentially perceive. Light output from the diode-pumped solid-state lasers is modulated according to the input signal, and the RGB light is combined. This combined light is then raster-scanned onto the screen to create an image. The laser projects the image either onto a flat area or directly onto the retina. Figure 8.15 shows the principle behind a retinal scanning laser display (RSD).

A collimated low-power laser diode is normally used as a modulated light source. The tiny laser beam is subsequently deflected in the u- and v-direction by two uniaxial scanners. The horizontal scanner (u-direction) operates at several kilohertz. The vertical scanner frequency (v-direction) defines the image refresh rate, which must exceed the critical

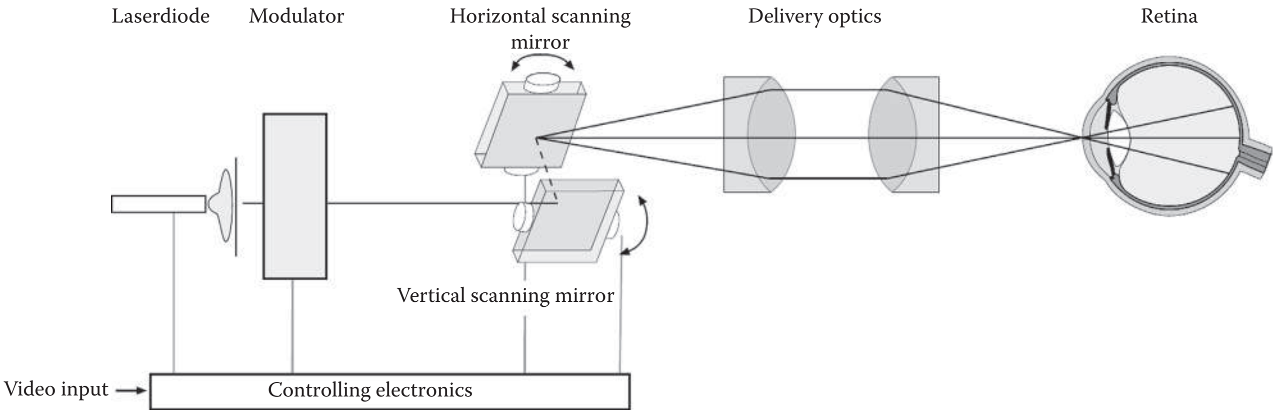


FIGURE 8.15 Illustration of the retinal scanning principle with dual scanning mirrors. (Adapted from Schowengerdt, B. T., and E. J. Seibel. 2006. *J Soc Inf Disp* 14[135]:139.

fusion frequency of about 60 Hz to prevent flickering effects. Finally, a viewing optic projects the laser beam through the center of the eye pupil and onto the retina. The RSD system projects pixels onto the retina as a serial sequence. Because the image refresh rate is more than the temporal resolution limit of the human eye, the user does not perceive any flickering effects (von Waldkirch 2004). Note: While all commercial projection technologies are considered "eye safe" by international standards, these standards distinguish varying levels of eye safety. Right now scanning-mirror technologies are rated Class 2-safe because the normal blink response limits exposure.

8.4.8 Other Nonemissive Displays

For nearly 2000 years, the most popular way to display words and images was to use ink on paper. It still has many advantages over electronic information displays in terms of legibility and price. The biggest limitation of paper displays is that the printed symbols cannot be changed or removed without leaving noticeable marks. The invention of electronic ink has made it possible to overcome this disadvantage. Electronic ink displays are based on the electrophoretic principle. These displays are reflective and promise paper-like properties, that is, wide viewing angle, thin, very flexible, and relatively inexpensive. Unlike paper, they are electrically writeable and erasable. One big advantage over other types of displays is their low-power consumption, which may extend the battery life of devices with such displays to months or even years.

Several nonemitter display technologies such as electronic ink, TFT LCD, and LCoS (see Section 8.4.3) are already on the market, whereas others are still in the fundamental development stage. Among them, the above-mentioned electrophoretic displays as well as electrowetting, cholesteric, and mirasol displays are especially promising for innovative commercial off-the-shelf products. 3QI Multimode, electrofluidic pixels, and photonic ink are generally promising but not as far developed (Heikenfeld 2010).

Electrophoretic displays: The electrophoretic principle was developed by Joseph Jacobson in the early 1990s. Jacobson cofounded the E-Ink Corporation, which later coproduced the first market-ready electrophoretic display (electronic paper) with Phillips Components (Castillo 2010). Electrophoretic displays generate images, viewable in reflective light, by rearranging laden pigment particles using an applied electric field. Titanium dioxide particles are dispersed in a hydrocarbon oil or water. The particles are about 1 μm in diameter. The suspension is allocated between two parallel conductive plates that are 10–100 μm apart (Figure 8.16). Applying voltage over the two plates stimulates the particles to migrate electrophoretically to the plate bearing the opposite charge. The screen appears white when white particles are at the front of the display, the viewing side. Light is scattered back to the observer in the color of the high-index titanium particles. When the particles are electrically stimulated to the rear side, the screen appears dark because the oil film now covers the viewing side. Dividing the rear electrode into a chosen number of small elements (pixels) allows a desired image to be generated in accordance with the resolution of the now-formed screen. An appropriate voltage must be applied to each region in order to create a pattern of dark and light areas.

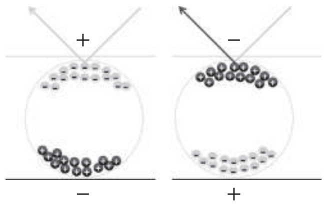


FIGURE 8.16 Illustration of electrophoretic microcapsules with adjacent electric field.

Electrowetting displays: As with electrophoresis, electric voltage stimulates the generation of an image in a suspension of oil and water with microcapsules. The oil (colored) forms a smooth layer between the water and a water repellent when no voltage is applied. This insulates the coating on the electrodes, resulting in a colored pixel. The tension changes between the water and the coating when voltage is applied between the electrode and the water, and the water moves the oil away. The result is a partially transparent pixel. If a white surface is used under the switchable element, the observer sees a white pixel. Due to the small size of the pixel, the user only perceives an average reflection. This means that a switchable element is generated possessing high brightness and high contrast. The element may then form the basis of a reflective display. Electrowetting displays have many advantages: the switch rate allows video, it is a low power and low voltage technology, and screens based on this technology may be very flat and thin. In addition, reflective characteristics are superior or at least equal to those of other reflective technologies, full-color displays with a relatively high brightness are possible, and each subpixel is able to switch independently between two different colors instead of using the three fundamental colors as a filter or alternating between segments of the three. This means two thirds of the display may reflect light in any color. This is achieved by constructing a pixel with a pair of two independently assignable colored oil films and a color filter. The colors used are cyan, magenta, and yellow. This system does not need any polarizers, meaning that e-wetting displays can be twice as bright as ordinary LCDs (Sun and Heikenfeld 2008).

Cholesteric LCDs: A cholesteric liquid crystal is a type of liquid crystal with a helical structure. Cholesteric liquid crystals are also known as chiral nematic liquid crystals. They appear in layers with no positional ordering within the layers, but they do have a director axis that varies with each layer. The variation of the director axis tends to be periodic in nature. The period of this variation (the distance over which a full rotation of 360° is completed) is known as the pitch. The pitch varies with temperature and can also be affected by the boundary conditions when the chiral nematic liquid crystal is sandwiched between two substrate planes (Yeh 2009).

Mirasol displays: Mirasol displays are based on interferometric modulator display (iMoD) technology. The iMoD was invented by Mark W. Miles, a microelectromechanical systems pioneer (Miles 2004). The technology consists of an electrically switched light modulator with a micromachined cavity that is operated using driver ICs similar to those used to address LCDs. An iMoD reflective flat panel display may include hundreds of thousands of individually assignable iMoD elements. iMoD displays incorporate the foremost examples of microelectromechanical systems based devices. An iMoD either reflects light at a given wavelength and the resulting light is seen as pure, bright colors, or it may appear black to the viewer because it can absorb all incident light on its surface (Mphepo et al. 2010).

3QI Multimode: This technology from Pixel Qi is a hybrid system which combines reflective and transmissive characteristics. It enables comparatively low-power consumption and indoor and outdoor viewing. However, compromises had to be made in terms of brightness and color saturation. Several reviews suggest that this technology may appear soon in laptop screens. The system combines a standard LCD with a standard black and white electrophoretic display and a restricted color electrophoretic display (Heikenfeld 2010).

Electrofluidic pixels are small reservoirs that may be filled with an ink-like fluid. An applied voltage either fills or empties the reservoir with an ink, generating the picture. The switch rate is very fast and allows the display of videos but also reaches paper-like conditions, making it very comfortable for the observer to read text (Heikenfeld et al. 2009).

Unlike the first prototypes of electronic paper, which could only display two colors, usually black and white, photonic ink can display any color value in the spectrum. Photonics (from photon) is an area of study based on the utilization of radiant energy, such as light, for various applications (Duncan 2008). Photonic ink, or p-ink, consists of planar arrays of silica microspheres—an opal film embedded in a matrix of cross-linked polyferrocenylsilane. Photonic ink is able to display color by controlled Bragg diffraction of light. Varying the size of the spaces between the particles creates different colors. A polymer gel is filled between the stacked spheres. The gel swells when soaked in a solvent and shrinks when it dries out, changing the distance between the tiny spheres. Because the size of the spaces determines the wavelength that is reflected, the swelling and drying of the gel results in a continuously tunable display of color across the entire visible spectral range. The amount of solvent absorbed by the gel is controlled by applying an electrical voltage. The optical response time of the film to a change in solvent is less than half a second (American Chemical Society 2003).

Electronic paper is often regarded as a substitute for paper products such as books and newspapers, as rewriteable paper in the office, as a material for price tags and retail signs, as wall-sized displays, and as a low-power display for smartphones. The general demand of the consumer for information "on the move" by means of sufficiently large, mobile, robust displays has consistently increased over the past years (Bhowmik, Li, and Bos 2008). New technological concepts to satisfy these demands include foldable and flexible displays that can be wrapped around a device, making the screen size to scale with the actual surface area (Myeon-Cheon, Youngkyoo, and Chang-Sik 2008). These "bendable display devices" can have a thickness of only a few micrometers. Candidate technologies are electrophoretic, cholesteric, electrowetting, and OLED displays. Except for OLEDs, all those technologies are reflective so they have certain limitations and are produced only in small scale (Crawford 2005).

8.4.9 Touch Sensitive Displays

Since their introduction the 1970s, touchscreens have turned from a futuristic input medium into a commodity device. The appearance has become so common that people today often don't take any extraordinary notice of this versatile technology. The use of touchscreens varies from mobile phones to desktop computers, handheld tablets, and large presentation screens. Elderly and disabled users in particular prefer touchscreens because of their intuitive direct manipulation mode (Mertens et al. 2010; Schneider et al. 2007).

The most popular technologies for touchscreens are as follows.

Resistive touch screens react to pressure generated by a finger or stylus. A resistive touch screen principally consists of several layers. Two of these layers are electrically conductive and slightly separated. Pressure "connects" these two layers and they act as voltage dividers generating two output currents. This change is identified as an input signal and is processed further according to the selected device drivers.

Capacitive touch screens consist of an insulator coated with a transparent conductor. Generally glass serves as the insulator. The human body is a natural conductor and is also used as such in this technology. A touch on the screen's surface results in an alteration of the human body's electrostatic field, which may then be measured as a change in capacitance.

Surface acoustic wave touch screens are widely used in publicly accessible interfaces such as ATMs and ticket vending machines. The system uses the ultrasonic wave created by a fingertip on the surface, which in this case may be a plain glass plate. Sensors aligned on every side (left, right, top, bottom) emit a sonic wave and also detect the wave emitted from the other side. When a finger makes contact with the screen surface, the sonic waves emitted are absorbed through the finger and the sensor arrays may locate the fingers position according to the interference patterns.

Optical touch screens use several optical sensors (such as infrared sensors) around the corners of the screen. The sensors are located in the range of a camera across the screen, such as on a wall on the other side of the display. A touch appears as a shadow on the screen and optical sensors may apply a simple triangulation algorithm to identify the location of the movement or touch.

Dispersive signal touch screens detect the mechanical load created by a touch. The general advantage of this technology is its relative resistance toward outside influences. Even scratches on the screen surface allow a further use of the system.

Strain gauge touch screens, also known as force panel technology, are spring mounted at every corner. Strain gauge touch screen indentifies the corresponding deflection when the screen is touched and locates it. It may also detect a person's force and movement along the *z*-axis.

The ability of several touch screen technologies to serve multitouch applications finds particular demand in cooperative product design and project management. Multitouch screens allow the precise measurement of more than one touch point on screen simultaneously. They can therefore identify complex gestures for advanced interaction. From the technologies just mentioned, the following can be used for multitouch purposes: capacitive screens, optical (infrared) screens, and more recently, resistive screens. More details can be found in Brown (2008), Chen, Cranton, and Fihn (2011), Jhuo, Wu, and Hu (2009), Maxwell (2007), Mertens et al. (2011), and Saffer (2008).

8.4.10 3D Displays

In general, two kinds of 3D displays must be distinguished because the mechanisms that carry the depth information by the light field they produce differ significantly. Stereoscopic displays produce only the light fields of two plain images of different views on a scene. The light fields of both views are independent and they carry by themselves no depth information other than perspective and occlusion. In contrast, volumetric displays produce one homogenous light field that simulates the light field of real objects. This homogenous light field carries inherently multiple views on the object and the depth information resulting from different focal lengths of object distances. Figure 8.17 shows how these two kinds of 3D displays can be further categorized according to technical principles.

Stereoscopic displays can be further distinguished according to the manner in which the images are multiplexed for the left and the right eye. Both images may be multiplexed spatially, temporally, or spectrally. For instance, shutter glasses multiplex both images temporarily. The display alternates images between the left and right eye. A synchronized shutter glass alternately blocks each eye's view so each one only perceives the image intended for it. In the case of polarizing glasses, different light polarization modes are used to encode and filter the images for the left eye and the right eye, respectively. Light can be polarized linearly or circularly. The modes of circularly polarized light are clockwise and counter-clockwise. The modes of linearly polarized light differ with the orientation of the plane along which they are polarized. The planes for the modes of linear polarization are perpendicular to each other. Using circular polarized light has the advantage that both images are correctly inverse multiplexed for arbitrary orientations of the user's head to the display screen, which is not the case if linear polarized light is used. Both images can also be multiplexed by using nonoverlapping spectral distributions for both images. One

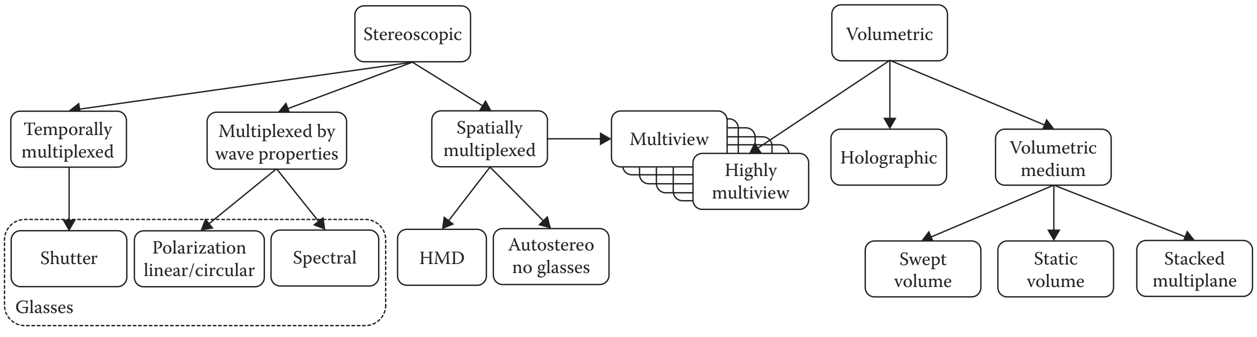


FIGURE 8.17 Taxonomy of three-dimensional displays.

example of this technology is an anaglyph that is viewed with two color glasses, where each lens has a chromatically opposite color, usually red and cyan. The disadvantage is that color perception is distorted. However, there are also more sophisticated filters in use that divide the visible color spectrum into six narrow bands—two in the red region, two in the green region, and two in the blue region (called R1, R2, G1, G2, B1, and B2 for the purposes of this description). The R1, G1, and B1 bands are used for one eye image, and R2, G2, and B2 for the other eye. The human eye is largely insensitive to such fine spectral differences so this technique is able to generate full-color 3D images with only slight color differences between the two eyes (Jorke and Fritz 2006). All these technologies have in common that the user needs glasses to see the 3D image. Autostereoscopic displays enable the user to see both images separately without using glasses. This is achieved by multiplexing both images spatially with an optical system that deflects the light of each image to the spatial position of the addressed eye. This is usually done by lenticular lenses in front of a common display and alternate columns in the image are deflected horizontally into different directions. This creates zones in front of the display in which one eye can only see one of the two images. Using lenticular lenses means halving the horizontal image resolution for each eye. New technologies combine temporal multiplexing with spatial multiplexing to implement autostereoscopic displays with full resolution per eye (Stolle et al. 2008). If the optical system that deflects the optical path of both images is not adapted to the movement of the user's head, the user may only see a clear 3D image within a restricted area in front of the display called the sweet spot.

One limitation of stereoscopic displays is that they do not support depth cues from motion parallax per se. This limitation can be bypassed if the position of the user's head is tracked and the generated images are computed to show intermediate views on the object according to the path of the movement of the user's head (Cruz-Neira, Sandin, and DeFanti 1993; Kruger et al. 1995).

Other autostereoscopic displays not only deflect two different images into different regions, but are able to deflect multiple views of an object in appropriate regions of visual space so that a user who is moving in front of the display has a stereoscopic view as well as depth cues from motion parallax between different views. If the angular density of views is sufficiently high, a single eye receives images of multiple views on the virtual object and it needs to adjust to the displayed depth of the object to see it clearly (Honda 2000; Takaki and Nago 2010). Thus, the light field becomes similiar to the light field of a volumetric display. In this sense, highly multiview displays can also be considered volumetric. Holographic displays are similar. They reconstruct the light field not by synthesizing multiple views through deflection but by scattering a chromatic wave front at fringe patterns. The scattered planar wave front reconstructs the original light field of a 3D object through interference (Leister et al. 2008).

Classic volumetric 3D displays use some medium in which dedicated voxels can be activated to emit light. Each voxel has a unique position (*x*, *y*, *z*) and can be assigned a color or gray scale value. Depending on the technology used to activate the voxels, volumetric displays can be assigned to several categories. Swept volume displays produce volume-filling images by reflecting or transmitting light from a rotating 2D surface within the desired 3D space. To display a point light at a specific location in the volume, it simply needs to be accomplished that any time the rotating 2D surface surpasses this volume element, it emits light of the desired color and brightness. This can be accomplished either by using active light elements like LEDs on the surface (Budinger 1984) or by using a laser that illuminates the point on a reflective surface at the right time (Langhans et al. 1998). Because of the human persistence of vision, an observer will perceive a stable 3D image if the volume is refreshed frequently enough. In a static volume display, there are no moving parts within the display volume. Several static-volume volumetric 3D displays achieve this by using a laser beam to excite visible radiation in a solid, liquid, or gas. Downing et al. (1996) developed a three-color volumentric display using high-power infrared laser beams. RGB voxels were created by sequential two-step absorption induced at the intersection of two laser beams with different wavelengths. The 3D image was drawn by scanning the volume with the laser beams to produce intersections at different locations inside the volume. Another technology to avoid moving parts is to use multi-stacked planes that can switch between a transparent and a light scattering state. Only one plane is in a light-scattering state, whereas the other planes are transparent. A high-speed digital projector reconstructs the scene by illuminating the voxels that lie within the successive opaque planes. A more detailed description of volumetric display technology can be found in Blundel and Schwarz (2000).

8.5 VISUAL PERFORMANCE

Impairment-free usage and user acceptance of interactive media depend substantially on the quality of the visual display as well as on the ease with which the displays allow information to be perceived and processed. A careful ergonomic evaluation is therefore essential to assess visual performance and identify existing shortcomings in displays.

Looking back, there is quite a long history of studies concerned with the ergonomic evaluation of visual displays (Dillon 1992; Schlick et al. 1997; Luczak and Oehme 2002; Pfendler and Schlick 2007). To estimate the costs and benefits of electronic displays in terms of human performance and visual load, the main interest was to learn which factors affect visual performance to what extent. One research approach was to compare display types in terms of visual processing efficiency. Most of these studies performed a basic comparison of the traditional hard copy with different electronic displays (Heppner et al. 1985; Gould et al. 1987). Another approach was to study the effects of specific quality criteria on performance (Pfendler, Widdel, and Schlick 2005; Sheedy, Subbaram, and Hayes 2003; Plainis and Murray 2000; Ziefle 1998, 2009). This procedure helps to distinguish different sources that account for performance shifts. Other studies gave attention to user characteristics (Kothiyal and Tettey 2001; Ziefle 2003a), workplace settings (Sommerich, Joines, and Psihoios 2001; Ziefle 2003b), and body posture (Aarås et al. 1997, 1998; Helander, Little, and Drury 2000) in the context of visual performance. Increasingly, ergonomic issues are under study for handheld devices and laptop computers, given the impact of reflection characteristics in TFT LCD displays, which often have disturbing glare and poor legibility under high levels of enviromental illuminance (e.g., Kubota 1997; Ziefle 2010a; Zingale, Ahlstrom, and Kudrick 2005).

Beyond display characteristics, any visual evaluation should also consider different task demands because they can strongly influence performance outcomes. The influence can be either positive, by compensating other suboptimal visual boundary conditions, or negative, when several negative factors accumulate, possibly leading to visual complaints and vision impairments.

8.5.1 Task Demands

Task demands are of crucial importance for evaluations of visual displays because they represent a complex entity of different factors and their interactions. The combination of (1) the task type (what the user is requested to do), (2) user characteristics (level of expertise, visual abilities, or working motivation), (3) text factors (font size, line pitch), (4) display factors (e.g., contrast, response times, resolution), (5) surrounding factors (e.g., ambient light or time on task), and (6) information access in multitasking settings has a considerable impact on performance outcomes. Because these factors were found to interact, it is necessary to examine the particular conditions and settings that were used in the different studies concerned with visual evaluations.

In order to evaluate visual displays, simple detection tasks, memory and recognition tasks, visual search tasks, and proof reading were used. Overall, two basic task forms can be distinguished: tasks in which a semantic context is present (e.g., proof reading) and tasks with no semantic context (e.g., visual search for multiple targets). This distinction has implications in terms of the ecological validity and generalizability of outcomes and in terms of the sensitivity with which shortcomings in visual displays are revealed. Tasks that have a semantic context have the general advantage of simulating what users usually do when using displays, thus the evaluation process is ecologically valid. However, these tasks were found be not very sensitive to visual degradation effects (Stone, Clarke, and Slater 1980; Ziefle 1998) for two main reasons. One is that the encoding and processing of text material represents behavior gained through intensive training. A top–down process comprehension guides the reader through the text and possibly masks the degradation effects. The second objection is concerned with reading strategies involving different combinations of cognitive and perceptual processes. When proofreading, for example, participants may read for comprehension or scan for unfamiliar letter clusters and word shapes. Thus, it is possible that performance outcomes do not reflect degradation properly even when display quality is suboptimal. Deteriorations might then be overlooked, especially in short test periods. Even though ecological validity is lower, visual search and detection tasks showed a higher sensitivity to degradation effects because the visual encoding process predominately relies on visual properties of the display (bottom-up process) without being masked by compensating cognitive strategies.

But text factors were also found to considerably affect performance. Among them, display size, the amount of information to be processed at one time and font size (Duncan and Humphreys 1989; Oehme et al. 2001; Ziefle, Oehme, and Luczak 2005; Ziefle 2008, 2010a) were revealed to be crucial in this context. The smaller the font size and the larger and denser the amount of (text) material to be processed, the stronger the performance decrements were. Age-related differences in response time can best be compensated by enlarging the font size from 16 to 22 arc minutes. Results from partial correlation analysis show that an age-differentiated adaptation of font size is recommended rather than an adaptation based on measurement of visual acuity (Vetter et al. 2010).

Whenever a detailed visual evaluation procedure is needed to analyze the effects of visual displays on performance, a two-step procedure is recommended. A benchmark procedure is advised as a first step. This procedure includes young and well-sighted participants and a task that is visually rather than cognitively strenuous. The second step includes a broadened scope. Older users (as they represent a major part of the workforce) and tasks with different visual and cognitive demands need to be addressed. In addition, extended periods of on-screen viewing are to be examined in order to realistically assess the long-term effects of computer displays.

8.5.2 Measures Used for Evaluation

The measures used to quantify the effects of visual display quality also differ regarding their sensitivity to visual degradation. In addition to effectiveness and efficiency of task processing, judgments of visual strain symptoms and the presence of visual complaints were also assessed.

Among performance measures, global and local parameters can be distinguished. Global performance measures are the speed and accuracy of task performance; they were widely used across visual evaluation studies. Reading and search times reflect the basic velocity of the encoding and visual processing. However, speed can only be meaningfully interpreted when the accuracy of task completion is also considered. In this context, the speed-accuracy tradeoff has a considerable impact. Whenever visual conditions, in computer monitors for example, are suboptimal (under low resolution or contrast conditions), the speed and accuracy of information processing cannot be simultaneously kept at a constantly high level. Rather, one of the two components deteriorates while the other remains constant. When both components are considered, the overall costs for information processing can be assessed. Nevertheless, reliable, global measures do not provide an understanding of the processes that cause the performance decrement. Oculomotor behavior was thus consulted to gain deeper insights into the nature of the deteriorated encoding process (Owens and Wolf-Kelley 1987; Iwasaki and Kurimoto 1988; Jaschinski, Bonacker, and Alshuth 1996; Best et al. 1996; Piccoli et al. 2001). The spatial and temporal characteristics of saccades were predominantly analyzed. Under visual degradation conditions, more saccades are executed, they are smaller in size, and they are accompanied by increased fixation times (Baccino 1999; Ziefle 1998, 2001a,b). Accommodation states, pupil size, vergence efforts, and visual scan path complexity (Schlick et al. 2006) were also used as measures to quantify the effects of display quality. The higher effort required for the visual system observed when reading electronic texts is assumed to form the physiological basis of visual fatiguing (Wilkins et al. 1984). Beyond oculomotor measures, stressor variables such as heart rate variability and eye blinks were used to determine the effects of display quality (Oehme, Schmidt, and Luczak 2003).

Another approach to quantifying effects of display quality is to collect user judgments with respect to what is called visual fatiguing, visual stress, visual strain, visual load, or discomfort. Visual fatiguing is considered a complex subjective measure based on the awareness of several symptoms: burning, dry, aching, watery eyes, difficulties in reading (text becomes blurred and fades away), as well as increased blinking and eye pressure (Piccoli et al. 2001; Stone, Clarke, and Slater 1980; Hung, Ciuffreda, and Semmlow 1986). User judgments represent an important aspect of display quality because perceived visual comfort is the most direct source of users' satisfaction (e.g., Oetjen and Ziefle 2007a,b, 2009). Even though ratings are easy to obtain, their validity is not without controversy. First, users differ considerably with respect to their responsiveness to visual strain symptoms. Furthermore, sensitivity to visual stress is not constant, but changes with the time spent on the task and age (Wolf and Schaffra 1964). Second, factors like fear of failing and the misinterpretation of one's own performance may contaminate judgments and make it necessary to prove that ratings match performance outcomes. Third, the emergence of visual fatiguing does not necessarily follow the same time course as performance shifts. Visual fatiguing is therefore not necessarily accompanied by performance decrements and vice versa (Chen and Lin 2004; Lin et al. 2009; Yeh and Wickens 1984; Howarth and Istance 1985). It is therefore advisable to include both subjective and objective measurements to obtain a complete evaluation.

8.5.3 Effects of Specific Quality Criteria on Visual Performance

As a first quality criterion, effects of display resolution (see Section 8.3.3) are considered. The psychophysical correlate of display resolution, and this is of visual ergonomic interest, is the sharpness of contours and the clarity with which objects can be identified. High-display resolutions allow more objects to be displayed on a given screen space. Even if this is advantageous when screen space restrictions are considered, it can be counter-productive: The benefit of higher resolutions in terms of contour sharpness may be negated by the smaller object sizes, and this has to be counterbalanced. The central question is whether higher resolutions lead to quantifiable improvements. The relevant literature shows that this is indeed the case (Gould et al. 1987; Huang, Rau, and Liu 2009; Miyao et al. 1989; Ziefle 1998). Due to unequal technical standards over time, resolution levels differ considerably between the studies (40–90 pixels per inch). In summary, it can be said that for up to 90 pixels per inch, higher resolution is better. Young adults' search performance was increased by 20% in the 90 pixels per inch condition compared with the 60 pixels per inch condition. This greater effectiveness was caused by a more efficient oculomotor control: Fixation times decreased by 11%, and in addition, less saccades (5%) were executed to process the visual information. Furthermore, performance in the low-resolution condition was not only found to be interrelated with the emergence and strength of visual fatiguing symptoms, but also the probability of fatiguing symptoms was significantly higher when two suboptimal viewing conditions were coincident: The longer participants worked in the low-resolution condition, the stronger performance decrements were (Huang, Rau, and Liu 2009; Ziefle 1998).

A second criterion to be examined extensively was intermittent light stimulation, which is characteristically present in CRT screens and is referred to as refresh rate (see Section 8.3.5). The perceptual component is a flicker sensation at low-refresh rates. Reading on a screen with low-refresh rates (50 Hz) is extremely hard work for users and leads to considerable eyestrain symptoms even after short reading periods. With increasing refresh rates (>70 Hz), the flicker sensation decreases, but it should be noted that the intermittent stimulation is nevertheless still physically present and may affect performance. It was shown that performance with a 100-Hz screen was 14% better than 50-Hz screen, and ocular efficiency was 16% better (fixation times and the number of fixations per line). However, performance decrements were also found at high-refresh rates (140 Hz), showing that effects of refresh rates on performance do not follow a linear, but a curvlinear relationship (Ziefle 2001a). In summary, we can say that refresh rates of 90–100 Hz in CRTs facilitate reasonably good visual performance.

Third, effects of luminance (see Section 8.3.2) and contrast (see Section 8.3.3) are also reported to play a major role in visual performance when computer displays are used (Plainis and Murray 2000; Van Schaik and Ling 2001; Ziefle, Gröger, and Sommer 2003; Oetjen, Ziefle, and Gröger 2005). However, results are not easy to integrate. Contrast is a complex factor, but it is also accompanied by several other lighting characteristics that are highly interdependent and affect the visual performance outcomes separately as well as in combination. Always a ratio of two luminance levels (background and object), in terms of Weber Contrast was proven to markedly affect the visual performance. Low contrast leads to performance decrements and oculomotor efficiency in a range of 10%–20%. But this ratio does not specify which absolute luminance levels constitute the respective contrast and—depending on which contrast definition is used—they mostly do not differentiate between whether information is displayed in negative or positive polarity (bright letters on a dark background vs. dark letters on a bright background). In addition, ambient lighting considerably interferes with the display contrast (Piccolo et al. 2004, 2007; Shen et al. 2009; Shieh and Lin 2000; Wang et al. 2009). Thus, knowing the contrast level without detailing polarity, absolute luminance levels and the kind of ambient lighting does not allow the interpretation of performance outcomes. Generally, it was found that positive polarity displays result in better visual performance than negative ones (Bauer and Cavonious 1980; Buchner, Mayr, and Brandt 2009; Wang and Chen 2000). The positive effect of positive polarity occurs due to the (simple) fact that light is essential for visual encoding (the absolute light level is higher with positive displays), and because it is less demanding when the computer display has the same polarity as the hardcopy (especially because both media are often used simultaneously in the same work setting) not requiring the user to alternately readapt. Ambient lighting usually reduces the contrast levels on the screen because the two lighting sources interfere. Thus, it is advantageous for performance if illumination in the room is sparse (Kokoschka and Haubner 1986). As for display luminance, it is recommended that luminance levels should not be too bright, preventing interfering glare, especially when ambient lighting is low and the probability of glare rises (Kubota 1997; Schenkmann, Fukunda, and Persson 1999).

Recent studies have shown renewed interest in the importance of contrast and luminance (Hollands et al. 2001, 2002; Gröger, Ziefle, and Sommer 2003; Ziefle, Gröger, and Sommer 2003; Oetjen and Ziefle 2004). Among visual factors, LCD displays (see Section 8.4.3) have the basic advantage of being flicker-free. However, they also have one major disadvantage: The displayed information is “perfectly” visible if users work in front of the screen and whenever this “optimal” position is not present, visibility is distinctly worse. This specific property of LCD screens is called “anisotropy.” A display is called anisotropic if it shows a deviation of more than 10% of its luminance subject to the target location or viewing angle (ISO 13406-2 2001). The nature of anisotropy is such that photometric measures (contrast and luminance) are not constant over the screen surface, but rather decrease with larger viewing angles. To quantify the change in photometric measures at different viewing angles, a measurement setup was developed which made it possible to exactly correlate photometric measures and visual performance (Gröger, Ziefle, and Sommer 2003; Ziefle, Gröger, and Sommer 2003). The screen was virtually cut into 63 fields (9 × 7). The luminance of bright/dark areas was individually measured by a photometer and contrasts were determined. Then different measuring positions were adopted: First, the “standard view” was applied, commonly used by the industry. The photometer was set in front of the screen and displaced gradually from field to field, with the photometer always set at right angles to the screen (Figure 8.18a).

From an ergonomic point of view, this procedure (as shown in Figure 8.18a) is highly artificial, as users do not displace themselves, but rather change their view: Viewing angles change remarkably depending on where users are looking. This fact is entirely disregarded in this measurement procedure. To simulate real viewing conditions, the “user view” (Figure 8.18b) and the “bystander view” (Figure 8.18c) were used. For the “user view,” the photometer was positioned centrally in front of the screen. Because the position of the photometer did not change, viewing angles increased with distance from the center of the screen, emulating a user’s head movements when looking toward the screen edges. For the “bystander view,” the photometer was set to a central point on the display and turned to the different measuring fields (30° and 50°). For extended viewing conditions,

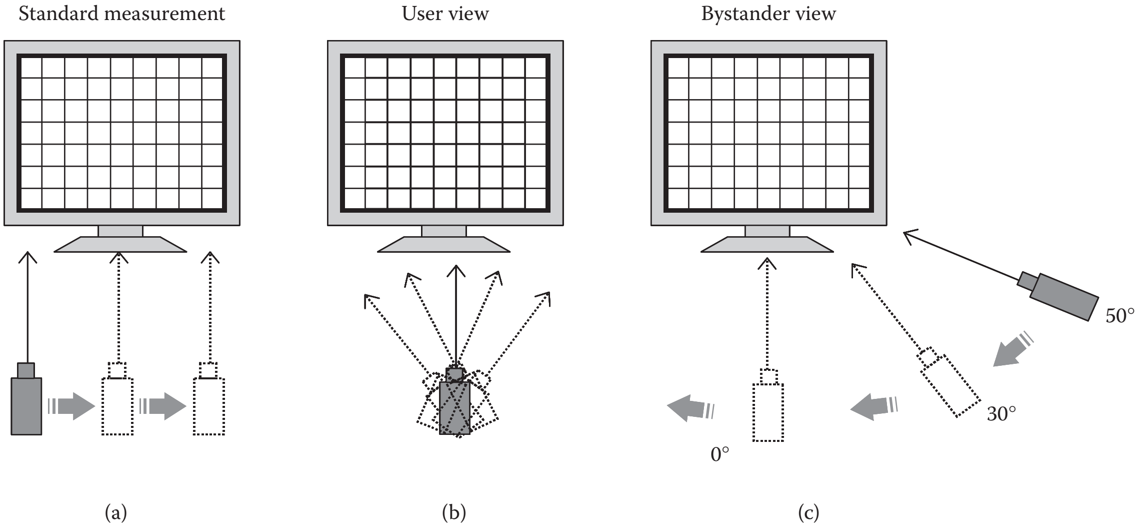


FIGURE 8.18 Quantifying anisotropy in LCD screens. (a) “Standard view” with the photometer displaced at right angles. (b) “User view” with the photometer emulating the user’s head movements. (c) “Bystander view” with the photometer positioned off-axis. (Data from Ziefle, M., T. Groger, and D. Sommer. 2003. *Int J Occup Saf Ergon* 9[4]:507–17; Oetjen, S., and M. Ziefle. 2009. *Appl Ergon* 40:69–81.)

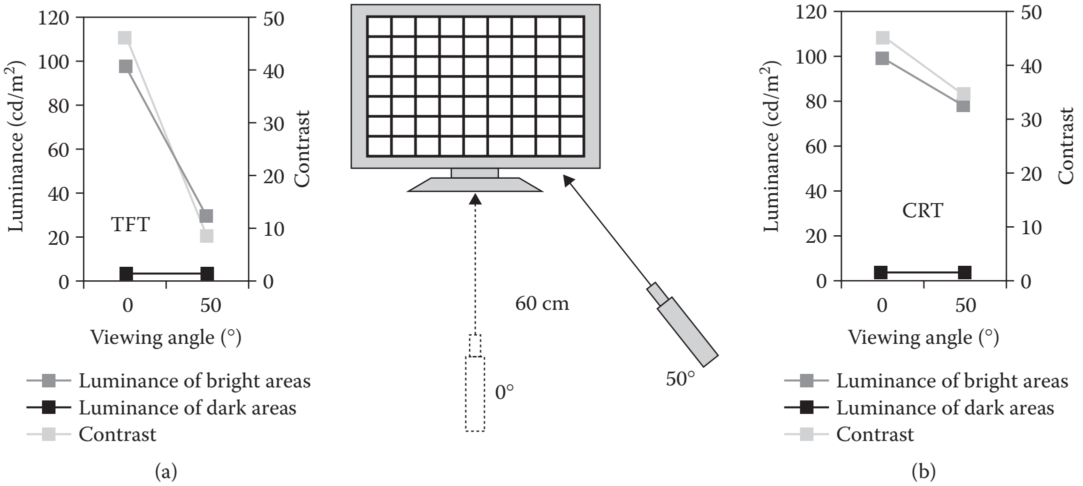


FIGURE 8.19 Photometric measures in the "bystander view" for (a) a liquid-crystal display and (b) a cathode ray tube display. (Data from Oetjen, S., and M. Ziefle. 2004. Effect of anisotropy on visual performance regarding different font sizes. In *Work with Computing Systems*, ed. H. Khalid, M. Helander, and A. Yeo, 442–7. Kuala Lumpur, Malaysia: Damai Sciences.)

the photometer was set off-axis, and its view pointed to the screen from the side (left and right side, respectively).

In Figure 8.19, it becomes evident that photometric measures change significantly as a function of viewing angle in LCDs (Figure 8.19a) and to a much lesser extent in CRTs (Figure 8.19b).

Performance outcomes showed that anisotropic effects have to be taken seriously. Visual performance with TFT LCD screens considerably deteriorates by about 10% for young adults when they had to look at the screen off-axis. Anisotropy plays a role in many real-life work settings: (air, rail) traffic controlling environments or stock exchanges use several displays simultaneously (placed next to and/or on top of each another) that have to be surveyed by one operator. Another example is the schooling and training context where it is quite normal for several users to work together on just one screen. Thus, from an ergonomic point of view, LCD's anisotropy must be regarded as a visual limitation of the display technology, at least when a fast and accurate visual detection performance is important.

8.5.4 Comparison of Display Types

Over the past 40 years, ever since the first evaluation studies of electronic information displays were published, a huge number of studies have dealt with the fundamental question of which display type assures the highest reading comfort and the best visual performance. Typically, and this reflects the chronological development, hardcopy, CRT, and LCD displays were compared for visual performance. As display quality has improved continuously over time, the technical standards on which the evaluations were based differ greatly. This is a factor that should be kept in mind. Recent studies include new developments (augmented reality, see Section 8.7.2) in visual display technology.

However, independently of the time and technical standard of visual displays, the one and only display that outperformed all others with respect to visual performance and comfort is the traditional hardcopy. Its development covers more than 200 years (since the invention of the industrial production of hardcopy in France) and has been continuously amended with respect to readability and visibility, by the expertise of typesetters and typographers. Thus, hardcopy can be regarded as an outstandingly suitable display with regard to visual ergonomic demands. It provides high contrast and resolution with neither disturbance by glare, screen reflections, or flicker. Accordingly, the majority of studies show that performance in CRT screens is significantly lower compared with paper (Heppner et al. 1985; Ziefle 1998). Hardcopy was also found to outperform modern TFT LCD displays (Ziefle 2001a).

Because the private and public need for electronically displayed information is continuously increasing nowadays, the evaluation should focus on the quality of different electronic displays. Here, the classic comparison of CRT and LCD displays is of central interest. The CRT, the most widespread screen type for decades, quickly phased out as a result of its specific lighting characteristics (flicker). The development of LCD technology was therefore highly welcome. LCD screens, lightweight and flat, are flicker-free and display information at much higher levels of luminance and contrast.

At first sight, the outcomes of studies that compare CRTs and LCDs reveal an inconsistent picture. There are studies in which TFT LCD displays led to higher visual performance than CRT screens (Menozzi et al. 2001; Ziefle 2001a,b), but there are also studies in which CRT displays outperform LCD screens (Oetjen and Ziefle 2004; Oetjen, Ziefle, and Gröger 2005; Oetjen and Ziefle 2007a,b, 2009). This inconsistency can be resolved when the experimental purposes and evaluation settings are considered. Whenever the impact of screen flicker was of main interest, the methodological focus was to compare both displays with respect to refresh rates (present in the CRT and absent in the LCD). This procedure ruled out any other differences between both displays that may confound or compensate flicker effects (anisotropy).

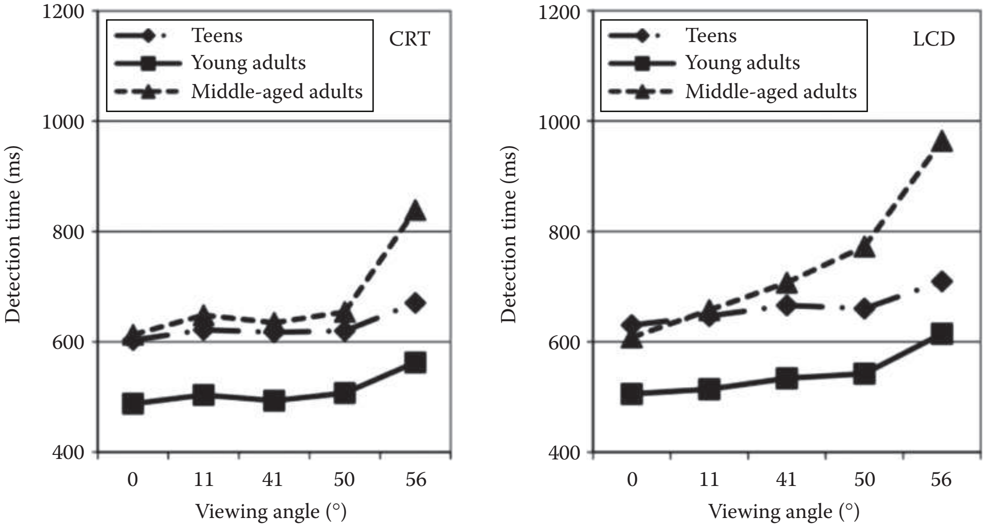


FIGURE 8.20 Effects of display type, viewing angle, and age groups on performance in a target-detection task. (Adapted from Oetjen, S., and M. Ziefle. 2007a. *Hum Factors* 49[4]:619–27.)

Thus, participants were usually seated in front of the screen and had to work on tasks that were displayed centrally (otherwise, anisotropic effects would have been mixed up). In these cases, visual performance clearly favors the TFT LCD and shows that screen flicker in CRTs is disadvantageous. In addition, it was found that the relative benefit from LCD technology is disproportionately higher for older users: While young adults (20–25 years of age) showed performance superiority of 10% for the LCD compared with the CRT (100 Hz), the benefit from the LCD was 16% and 27%, respectively, when older users (40–65 years of age) were examined (Ziefle 2001b). Note, however, that the older adults generally showed nearly 40% lower visual performance.

However, the benefit of the LCD determined in this way has yet to take anisotropic effects into account. As soon as anisotropy is considered, the picture changes. Studies concerned with the anisotropic effects of the two display types showed that the LCD's superiority over the CRT disappears when extended viewing angles were considered (Gröger, Ziefle, and Sommer 2003; Hollands et al. 2001, 2002; Oetjen and Ziefle 2004). When considering all screen positions, visual performance decreased by 8% when an LCD was used instead of a CRT. When a central view was applied, detection times were 14% faster than when viewing 50° off-axis. A further aggravating factor was font size. Deterioration when detecting small targets (1.5 mm vs. 2.4 mm) rose to almost 30% with the LCD, versus "only" 20% with the CRT. Whenever all suboptimal factors (small font, LCD screen, off-axis viewing) occurred simultaneously, performance decrements were found to rise to as much as 38% (Oetjen and Ziefle 2004, 2007a,b).

Visual ergonomic studies examined mostly young adults as participants. However, young adults do not represent the whole work force that uses electronic information displays. Rather, children and teenagers as well as older users do frequently use displays in private and professional settings. Until now, both major user groups have been mostly disregarded by visual ergonomic studies. Because visual functions change significantly with age (e.g., Ellemberg et al. 1999; Kline and Scialfa 1997), we need to know to what extent anisotropy affects visual performance in other age groups so that we can maximize work productivity, safety, and comfort. In one study (Oetjen and Ziefl 2007a,b) teenagers ($M = 13.9$ years), young adults ($M = 23.9$ years), and middle-aged adults ($M = 56.4$ years) were compared for visual performance when working with anisotropic displays (TFT LCD display (CF-L 15, 15-in., 1024 × 768), and a CRT as a control condition (Sony S 200 PS, 17-in., 100 Hz, 1024 × 768). Five different viewing angles were studied (0°, 11°, 41°, 50°, and 56° off-axis). Speed and accuracy of visual performance in a simple detection task was measured. The results show a significant impact of anisotropy on performance. Detection times were 7.6% longer for the LCD compared with the CRT and increased by 21.9% from the central (0°) to the off-axis conditions (56°). The LCD's anisotropy does not equally disadvantage all age groups, but rather appears to be age-related (Ziefle 2009). This is shown in Figure 8.20.

Laptop computers are replacing stationary desktop systems more and more because of increasing demands for work mobility (Kirsch 2004). Extended viewing angles are especially important in mobile contexts. In order to analyze the impact of LCD's anisotropy, visual performance in off-axis viewing conditions was investigated, comparing a mobile TFT LCD to a stationary LCD and a CRT (Oetjen and Ziefle 2009; Ziefle 2009). The study simulated real work situations where several users are viewing one screen or one "operator" has to survey several screens at the same time. Participants worked from two sitting positions (central position where the user was placed directly in front of the screen and an off-axis position where the user was placed off-axis). Again, five different viewing angles were examined (0°, 11°, 41°, 50°, and 56° off-axis). In accordance with findings of earlier studies (e.g., Gröger, Ziefle, and Sommer 2003; Hollands et al. 2002; Oetjen and Ziefle 2004, 2007a,b; Ziefle, Gröger, and Sommer 2003), the present study corroborated anisotropy as a major shortcoming of LCD displays. Although LCD displays have

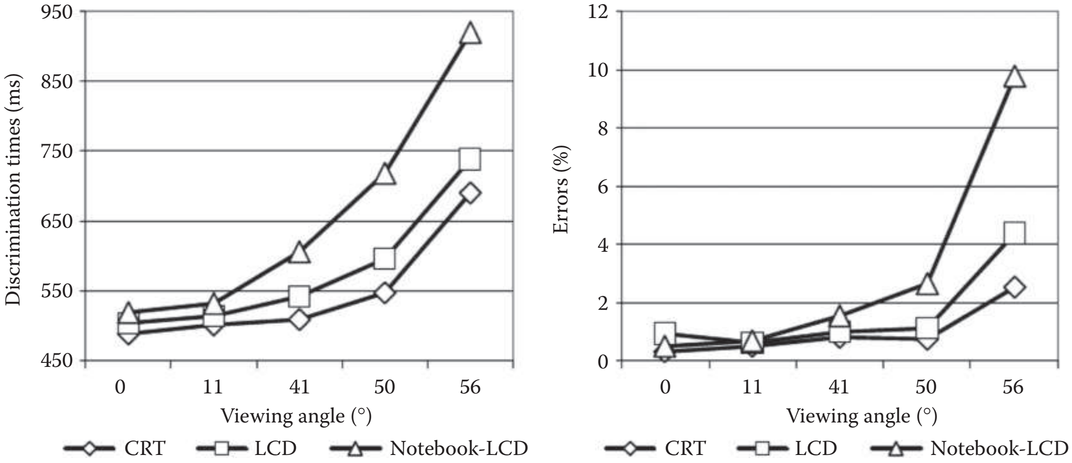


FIGURE 8.21 Visual performance in terms of target discrimination times under different off-axis viewing conditions for three display types. (Adapted from Oetjen, S., and M. Ziefle. 2009. *Appl Ergon* 40:69–81; Ziefle, M. 2009. Visual ergonomic issues in LCD-Displays. An insight into working conditions and user characteristics. In *Methods and Tools of Industrial Engineering and Ergonomics for Engineering Design, Production, and Service-Traditions, Trends and Vision*, ed. C. M. Schlick, 561–72. Berlin: Springer.)

many advantages (see Section 8.4.3), one disadvantage is that luminance is not isotropic, but varies as a function of viewing angle. Physical measurements revealed the strongest fluctuation of luminance for laptop TFT LCDs, followed by desktop LCDs (Oetjen and Ziefle 2009). The CRT technology, however, is not significantly affected by anisotropy. Performance data mirror these differences although the extent of performance deterioration is smaller than the reduction of luminance levels (Figure 8.21). When the discrimination performance of all screen positions is considered, using the CRT led to the best performance and using the laptop LCD led to the worst performance. In all screen positions, the mean difference was about 6% when CRT and external LCD were compared, and it increased to a surprising 18% between CRT and laptop LCD. The strong susceptibility to off-axis viewing becomes even more evident when only the off-axis conditions are considered. At the 56° position, the speed of visual discrimination decreased by 33% when the laptop LCD was compared with the traditional CRT.

In summary, LCD anisotropy is a limiting factor for visual performance, especially in work settings where fast, accurate reactions are necessary. Nevertheless, it should be taken into account that any comparison of laptop LCDs and external LCDs is unbalanced if only one dimension—visual quality—is focused upon. Other aspects of working contexts are also important in real-life applications. One must consider that the major advantage of laptop computers is their ability to be mobile and change work settings. Besides their susceptibility to the negative effects of restricted viewing angles, it should also be mentioned that for privacy reasons the effects are sometimes highly welcome (e.g., in ATMs and mobile phones).

Recommendations for display types should therefore be related to the specific task context in which they are to be used. When considering human performance in off-axis viewing conditions from a visual ergonomic perspective, there is a clear ranking of screen types for this kind of task demand. Considerably lower performance is to be expected when LCDs are used, when time-critical tasks have to be completed, when the whole display surface is used to display the stimuli and when extended viewing angles are present. The decrements are most pronounced in laptop LCDs because their LCD technology is very susceptible to off-axis viewing conditions.

8.6 STANDARDS, REGULATIONS, AND SEALS OF APPROVAL

There are a large number of national and international standards, regulations, and seals of approval that regulate multiple aspects of the design and use of visual displays. They cover areas such as ergonomics, emissions, energy consumption, electrical safety, and documentation. Various federal institutions and private organizations such as the European Agency for Safety and Health at Work, the German Technical Surveillance Association (TÜV), or the American National Standards Institute (ANSI) are responsible for product surveillance and awarding safety seals and recommendations for use. Similarly, the ASEAN Consultative Committee on Standards and Quality (ACCSQ), which represents 10 Southeast Asian nations, aims to form a region of product standards. China and Japan have their own respective federal institutions, SAC and JAS. Display users should generally be aware that many of the awarded quality and ecological seals are often created by the producer and therefore do not guarantee a rigorous quality comparison among different producers and their products. The significant rise in the importance of ecological labeling and consumer demands has prompted producers to join in this widespread global industrial trend. Public awareness of this matter has led the industry to highly praise their own products and advertise them with attention to their ecological thriftiness (e.g., "standby" energy balance of visual displays).

ISO Technical Committee 159 (ISO/TC 159 SC 4/WG 2) is an important international committee for the development of ergonomics standards for visual displays. The most important directives for all display types can be found in Table 8.1.

TABLE 8.1
Selected List of National and International Standards, Regulations, and Seals of Approval Regulating Design and Use of Visual Displays

	Ergonomics
Image quality	DIN EN 29241-3/A1, ISO 13406-2, TCO'06, GS Mark, Ergonomics Approved Mark, ISO/IEC 24755
Reflection characteristics	DIN EN ISO 9241-7, ISO 13406-2, TCO'06, GS Mark, Ergonomics Approved Mark
Color requirements	DIN EN ISO 9241-8, ISO 13406-2, TCO'06, GS Mark, Ergonomics Approved Mark
Brightness and contrast adjustable	European VDU directive 90/270 EEC, German ordinance for work with visual display units, TCO'06, GS Mark, Ergonomics Approved Mark
Tilt and swivel	TCO'06, GS Mark, German ordinance for work with visual display units
Operation	DIN EN ISO 11064-5, ISO/IEC 11581-1, ISO/IEC 18036
Gloss of housing	GS Mark, German ordinance for work with visual display units
	Testing and evaluation
	ISO 9241-304, ISO 9241-306
	Emissions
Noise	European VDU directive 90/270 EEC, ISO 7779/A1 (ISO 9296) German ordinance for work with visual display units, TCO'06, GS Mark
Electrostatic potential, electrical, and magnetic fields	PrEN 50279, TCO'06, Ergonomics Approved Mark
X-ray radiation	TCO'06
	Energy consumption
	EPA Energy Star, TCO'06, VESA DPMS
	Electrical safety
	TCO'06, GS Mark
	Documentation
Technical documentation and user manual	German Equipment Safety Law, GS Mark

Due to rapid technological development, only selected criteria can be covered. Up-to-date information and advanced information can be found on the websites listed in Table 8.2.

8.7 DISPLAYS FOR SELECTED APPLICATIONS

8.7.1 Virtual Reality

Immersive VR is a technology that enables users to "enter into" computer generated 3D environments and interact with them. VR technology involves the additional monitoring of body movements using tracking devices, enabling intuitive participation with and within the virtual world. Additional tracked peripheral devices permit virtual navigation, pick-and-place manipulation of virtual objects (Schlick, Reuth, and Luczak 2000) and interaction with humanoids and avatars by using data gloves, space joysticks, or 3D tracked balls (Holmes 2003).

A 3D view is generated using different concepts and technologies (see Sections 8.2.5, 8.3.7, and 8.4.10). HMDs are a commonly used display device. For VR, a closed-view HMD in non-see-through mode is usually used (see Section 8.7.2). Visual displays, especially HMDs, have decreased substantially in weight since the first invention of an immersive head worn display, but are still hindered by cumbersome designs, obstructive tethers, suboptimal resolution, and an insufficient field of view. Recent advantages in wearable computer displays, which can incorporate miniature TFT LCDs directly into conventional eyeglasses or helmets, should simplify ergonomic design and further reduce weight. Most of the advanced closed-view HMDs have adjustable interpupillary distance in order to avoid mismatches in depth perception. They provide a horizontal field of view of 30°–50° per eye and a resolution of at least 1024 × 768 and therefore outperform predecessor systems (Stanney and Zyda 2002). Large images of animated VR scenes can be generated by LCoS-based or DMD projectors on the front or back of one or multiple screens. Stereoscopic images are projected on the basis of spatial, temporal, or spectral multiplexing (see Section 8.4.10).

The computer animated virtual environment (CAVE) is a further development in projection technology. It consists of a cube with several panels onto which the images are projected from behind. Depending on the construction, there are C3 (two walls and the floor), C4, C5, and C6 designs. A CAVE provides space for small groups but can usually only track and optimize the stereoscopic view for one person. The other people perceive distortions, especially at corners and edges.

8.7.2 Augmented Reality

Augmented reality (AR) characterizes the visual fusion of 3D virtual objects into a 3D real environment in real time.

TABLE 8.2
Websites That Provide Information about National and International Standards, Regulations, and Seals of Approval for Visual Displays

	Websites
ISO	The International Organization for Standardization is a worldwide federation of national standards bodies from 140 countries. Internet: www.iso.ch
ANSI	The American National Standards Institute sets and monitors standards for the U.S. market. Internet: http://ansi.org
SAC	The Standardization Administration of China establishes standards for the import–export business. Internet: www.sac.gov.cn/templet/english
JSA	The Japanese Standards Association governs the directives for the Japanese market. Internet: www.jsa.or.jp/top.asp
ACCSQ	The ASEAN Consultative Committee on Standards and Quality brings together 10 Southeast Asian countries and their desire to form a common market with identical standards. Internet: www.aseansec.org/
EN	European standards are available on the website of the European Agency for Safety and Health at Work. Internet: http://europe.osha.eu.int/legislation/standards
Ecolabel	The European Commission label for environmentally friendly displays and other office devices. Internet: http://ec.europa.eu/environment/ecolabel/
DIN	German Institute for Standardization. Internet: www.din.de
TÜV Rheinland	The German Technical Surveillance Association sets standards for Germany's industrial and private sectors. The GS Mark shows conformity with the German Equipment Safety Law. The Ergonomics Approved Mark demonstrates that a visual display terminal complies with TÜV ergonomic standards. Internet: www.tuv.com
European directives	European directives regarding health and safety are available on the website of the European Agency for Safety and Health at Work. Internet: http://europe.osha.eu.int/legislation/
EPA	The U.S. Environmental Protection Agency (EPA) promotes the manufacturing and marketing of energy efficient office automation equipment with its Energy Star Program. Internet: www.energystar.gov/
National laws and ordinance	National laws, directives, and regulations regarding health and safety for many European and some other countries are available via the website of the European Agency for Safety and Health at Work. Internet: http://europe.osha.eu.int/legislation/
TCO	The Swedish Confederation of Professional Employees developed requirements for PCs. The TCO'06 label specifies ergonomic, ecological, energy consumption and emission requirements. Internet: www.tco.se
VESA	The Video Electronics Standards Association creates standards for transmissions between computers and video monitors that signal inactivity. Internet: www.vesa.org

Unlike virtual environments, AR supplements reality rather than completely replacing it (Azuma 2001). AR can be used in many applications, such as production (Schlick et al. 1997; Park et al. 2008; Odenthal et al. 2009), assembly and service (Kleiber and Alexander 2011), medical (Park, Schmidt, and Luczak 2005), architecture, entertainment and edutainment, military training, design, robotics, and telerobotics.

One approach to overlaying the real world with virtual information is to use an HMD (see Section 8.2.5). Superimposition can be done in two ways: using an HMD in see-through mode (optical-see-through) or an HMD in non-see-through mode, called video-see-through (feed-through). The HMD in the non-see-through mode optically isolates the user completely from the surrounding environment, and the system must use video cameras to obtain a view of the real world. The optical see-through HMD eliminates the video channel, so the user is directly looking at the real scene through optical combiners. The merging of the real work environment and virtual augmentation is performed by screen-based optical combiners in front of the eyes, often based on half-silvered mirrors. Both systems have advantages and disadvantages in usability and technology. In an optical see-through system, there is a time lag between the real-world information and the virtual information that is blended into the field of view. This is caused by the computing time for image generation. In addition, the calibration of such a system is rather complicated. In video see-through systems, the quality of the video depends on both the technology of the cameras and the displays. There are several factors such as limited response time, displacement of the cameras from eye level and time delay of the channels, which have an adverse effect on human perception and hand–eye coordination (Biocca and Rolland 1998; Oehme et al. 2001; Luczak et al. 2003; Park, Schmidt, and Luczak 2005; Ziefle, Oehme, and Luczak 2005). Another disadvantage is a certain loss of information caused by the fact that the perception of the real work environment is limited to the maximum resolution and field of view of the displays and cameras. An interesting approach to overlaying visual information is to use a VRD (see Section 8.4.7). Unlike the screen-based HMDs, a VRD reaches the retina directly with a single stream of pixels, thus guaranteeing a clear projection of different kinds of information. Because of the higher laser light intensity, the half-silvered mirrors commonly used for see-through HMDs can be optimized for maximum translucence, improving the see-through quality. Furthermore, the display's maximum resolution is no longer determined by the tolerances used in manufacturing the display pixels, but by the control logic and quality of the deflection mirror.

Ergonomic experiments based on geographic orientation tasks have shown that human performance in terms of task completion time and experienced task difficulty is significantly better when optical see-through HMDs are used instead of paper-based maps (Pfendler and Schlick 2007). Moreover, the evaluation of three different electronic information displays for geographic orientation tasks (optical see-through HMD with full-color AMOLED, full-colored TFT LCD hand-held display, and optical see-through HMD with two-colored VRD) showed no significant differences in human performance (Pfendler et al. 2011).

A recent study by Odenthal et al. (2009) investigated the ergonomic presentation of assembly information by an augmented vision system set up to support human operators in the task of detecting assembly errors in small work pieces. A high-resolution binocular/stereoscopic optical see-through HMD and a common monoscopic TFT LCD display, which was mounted on a table behind the work piece were used for visual augmentation. The statistical analysis showed that error detection rate with the HMD instead of the table-mounted display significantly increased. However, this was accompanied by a trend toward longer detection times.

A clip-on display tries to combine the optical and video see-through modes. A tiny TFT LCD screen can be clipped onto eyeglasses or safety glasses. The user of such a display can easily change between the generated image on the screen and the real world beyond it. However, the screen area covers a certain fraction of the field of view, so that objects can be fully or partially occluded.

8.7.3 Mobile Phones and Handheld Devices

Mobile information and communication technologies are one of the fastest growing technological fields ever and have interpenetrated many professional and private fields in the last decade. According to recent estimates, there are 4 billion users of Global System for Mobile Communications connections per year worldwide. Data are usually transmitted by mobile phones or smartphones. The main reason for their popularity is on-the-go lookup and entry of information, quick communication, and instant messaging (Weiss 2002). As mobile devices provide wireless Internet access, web services can be used everywhere and at any time (Rao and Minakaki 2003). Mobile computing has already expanded into very different areas such as field services, healthcare, and journalism. Experts predict that by 2013 more than 445 million people will be regularly using their mobile phones to purchase goods remotely (Informa Telecoms and Media Global 2008).

Mobile devices are all battery operated, and this imposes severe limitations on power use for the display. In contrast to early LCD displays, which were reflective and had a very limited resolution, today's emitter and nonemitter displays require a certain power budget to be dedicated solely to information visualization. Even with high-resolution passive-matrix displays, which are very power-efficient, an increase of nearly one order of magnitude in power consumption should be expected. The commercial release of Li^+ batteries provided a jump in power density and peak power, opening up the area to emitter and AM nonemitter displays. When using TFT LCD displays, power consumption can account for 80% of the overall system (Li, Bhowmik, and Bos 2008).

Handheld devices may differ with respect to physical dimensions, display resolution, contrast, luminance, and touch/stylus sensitivity, but they all are characterized by small screen size, something that has a considerable impact on human information processing. For instance, the number of menu items that can be displayed at a time on the screen is severely limited. Hence, many items must be memorized by the user. Some devices display as many menu items as possible to support information in a user's working memory, whereas others prefer to show only a few menu items per screen to improve legibility of characters. Current mobile phones (e.g., models from Nokia, Motorola, Samsung, Sony Ericsson) have screen sizes of about 3.5–6 cm (length) and 2.5–5 cm (width) and display between two and eight lines of characters. Character sizes vary between 2 and 5 mm. Display sizes for smartphones are typically bigger and have about a length of 6–9 cm and a width of 4–6.5 cm. Handheld devices are also often used in medical care so they must be small to meet intimacy and/or acceptance demands, for example, signal warning devices for blood pressure or diabetes (Calero Valdez et al. 2009, 2010; Mertens et al. 2009).

Limited screen space is very problematic for providing optimized information access, and the question of how to "best" present the information on the small display is challenging. At first glance, the challenge seems to be mainly related to visibility concerns. If that is the case, visually ergonomic principles should be given primary consideration. To provide fast, accurate information access, objects and letters should be big enough, text lines should not be too close together and information density should be low. This is especially important for older adults, who usually have problems with their sight (Brodie et al. 2003; Omori et al. 2002). However, visibility concerns are not the only point of concern. There is also the cognitive aspect of information visualization, that is, the requirement that the presentation of information should help users orient themselves properly. Disorientation in the menus of handheld devices is a rather frequent problem (Ziefle and Bay 2005, 2006; Ziefle 2008, 2010b). Users have to navigate through a complex menu of functions, which is mostly hidden from sight because the small window only allows a few functions or small text fragments to be displayed at a time.

Several techniques have been proposed to accommodate the problem of displaying a lot of information on a small screen. One is rapid serial visual presentation (RSVP), which is based on the idea of presenting information temporally instead of spatially (Rahman and Muter 1999; Goldstein et al. 2001). With RSVP, one or more words are presented at a time at a fixed location on the screen and users have to integrate text fragments bit by bit by scrolling the text forward and, if necessary, backward. A similar technique is the times square method (TSM), also known as leading. With TSM, the information is not static but moves across the screen (word by word or sentence by sentence), with the text scrolling autonomously from left to right. Even if trained readers reach a reasonable level of efficiency with both presentation modes (RSVP and TSM), users report a dislike for both presentation forms. The low acceptance may be because of high cognitive and visual demands imposed by the presentation mode. Either memory load is high because it is difficult for users not to lose the plot while integrating the words and sentences that are displayed one after another (RSVP), or visual and attentional demands are high because it is essential to catch the text content in moving sentences. Accordingly, when users fail to read content on the first attempt, they have to wait until the scrolling information appears again (TSM).

Considering visual and cognitive demands concurrently, two alternatives can be contrasted. The first alternative is to display only a little information on screen at a time. This helps avoid visibility problems resulting from high-information density. The other alternative is to display as much information on screen as possible. This allows users to have maximum foresight (cognitive preview) of other functions on the menu, which should benefit information access from a cognitive point of view and minimize disorientation. It appears that a sensitive cutoff needs to be defined between visual and cognitive impacts. Additionally, it needs to be determined whether the impact of cognitive preview and the impact of visual density are crucial for efficient information access. An initial study experimentally investigated the role of menu foresight (Bay and Ziefle 2004). Young adults processed tasks on a simulated mobile phone where one, three, or seven menu items were presented at a time. The results corroborated the significance of information presentation on small screens with regard to efficiency of usage: Intermediate foresight (three functions) was found to lead to the best performance. When only one menu item was shown at a time—as is the case for a number of devices on the market—users needed 40% more steps to process tasks than when three items were shown, conforming the cognitive facet to be crucial. But also when information density was high (seven functions), performance declined by more than 30% compared with the presentation of three menu functions per screen, confirming that the visual facet also plays an important role.

Another study (Ziefle 2010a) scrutinized the tradeoff between legibility and menu foresight (number of functions seen at one time). Ergonomically, this tradeoff is critical because displayed information needs to have a sufficiently large font size to provide good legibility. However, menu orientation is facilitated when the amount of information per screen is maximized and a large preview is allowed. Thus, it is of interest which of the two processes is critical for the usability of small screen devices—the influence of cognitive preview or the effects of legibility. Independent variables font size (8 pt, 12 pt), information density, and cognitive preview (one or five functions per screen at a time) were experimentally varied, and the effects on navigation performance in mobile phones were observed. Because older adults are an increasingly key target group for mobile technologies (Ziefle 2008, 2010b; Ziefle and Bay 2008), and because they sometimes have special problems with information representation on small screens, older participants were chosen to solve very common phone navigation tasks on a simulated mobile phone. Navigation performance was assessed according to task success (effectiveness) and time needed to solve phone tasks as well as disorientation measures, that is, the number of (unnecessary) returns in the menu hierarchy and the number of returns to the top menu level to start over. Both measures had previously been shown to be very sensitive to reflect problematic interface design and low usability (Ziefle and Bay 2006, 2008).

The outcomes clearly revealed that both visibility issues and orientation concerns have a major impact on older users' navigation performance. This can be taken from the fact that the best navigation performance—in terms of effectiveness and efficiency—was obtained for the display design with a large font size and a large preview. However, when weighing the relative impact of both factors, proper orientation in the menu is more decisive than visibility effects for the older group. The lowest performance resulted from the display with a large font size. Visibility there was good, but because the preview was small, it could only display one function per screen at a time (Figure 8.22).

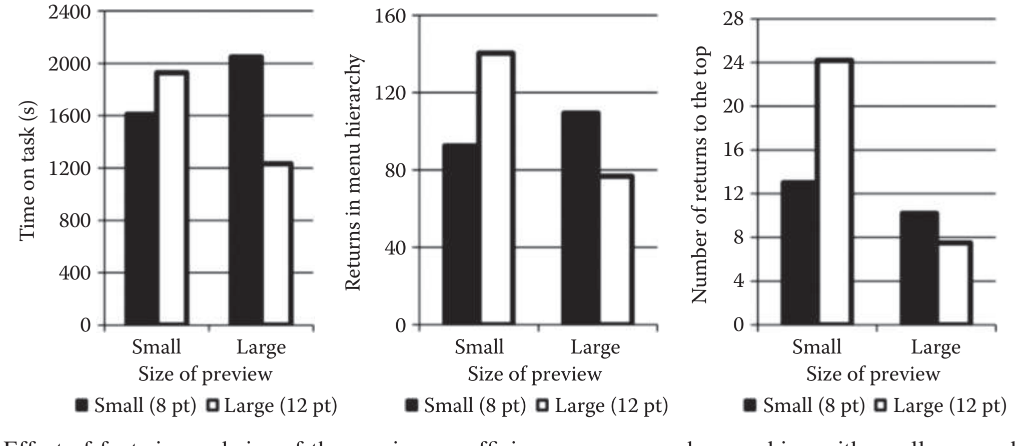


FIGURE 8.22 Effect of font size and size of the preview on efficiency measures when working with small screen devices. From left to right: time on task (s), number of returns in menu hierarchy, number of returns to the top. (Adapted from Ziefle, M. 2010a. *Applied Ergonomics* 41(6):719–30.)

REFERENCES

Aarås, A., K. Fostervold, O. Ro, M. Thoresen, and S. Larsen. 1997. Postural load during VDU work: A comparison between various work postures. *Ergonomics* 40(11):1255–68.

Aarås, A., G. Horgen, H.-H. Bjorset, O. Ro, and M. Thoresen. 1998. Musculosceletal, visual and psychosocial stress in VDU operators before and after multidisciplinary ergonomic interventions. *App Ergon* 29(5):335–54.

American Chemical Society. 2003. Multicolored Ink. http://pubs.acs.org/cen/topstory/8112/8112notw8.html (accessed October 1, 2005).

Amft, O., P. Lukowicz. 2009. From backpacks to smartphones: Past, present, and future of wearable computers. *J Pervasive Comput* 8(3):8–13.

Armbrüster, C., M. Wolter, T. Kuhlen, W. Spijkers, B. Fimm. 2008. Depth perception in virtual reality: Distance estimations in peri- and extrapersonal space. *Cyberpsychol Behav* 11:9–15.

Azuma, R. 1997. A survey of augmented reality. *Presence* 6(4):355–85.

Azuma, R. T. 2001. Augmented reality: Approaches and technical challenges. In *Fundamentals of Wearable Computers and Augmented Reality*, ed. W. Barfield and T. Caudell, 27–63. Mahwah, NJ: LEA.

Baccino, T. 1999. Exploring the flicker effect: the influence of in-flight pulsations on saccadic control. *Ophthalmol Physiol Opt* 19(3):266–73.

Bass, M. 1995. *Handbook of Optics*. Vol. 1. New York: McGraw-Hill.

Bauer, D., and C. R. Cavonius. 1980. Improving the legibility of visual display units through contrast reversal. In *Ergonomic Aspects of Visual Display Terminals*, ed. E. Grandjean and E. Vigliani, 137–42. London: Taylor & Francis.

Baumann, K., and H. Lanz. 1998. Mensch-Maschine-Schnittstellen elektronischer Geräte. In *Leitfaden für Design und Schaltungstechnik [Human-Machine-Interface for Electronic Appliances: Guideline for Design and Circuitry]*. Berlin: Springer.

Bay, S., and M. Ziefle. 2004. Effects of menu foresight on information access in small screen devices. In *48th Annual Meeting of the Human Factors and Ergonomic Society*, 1841–45. Santa Monica: Human Factors Society.

Best, P., M. Littleton, A. Gramopadhye, and R. Tyrell. 1996. Relations between individual differences in oculomotor resting states and visual inspecting performance. *Ergonomics* 39:35–40.

Bhowmik, A. K., Z. Li, and P. Bos. 2008. *Mobile Displays: Technology and Applications, Wiley Series in Display Technology*. West Sussex: John Wiley & Sons.

Biocca, F. A., and J. P. Rolland. 1998. Virtual eyes can rearrange your body: Adaption to visual displacement in see-through, head-mounted displays. *Presence* 7(3):262–77.

Blundel, B. G., and A. J. Schwarz. 2000. *Volumetric Three-Dimensional Display Systems*. New York: John Wiley & Sons.

Boff, K. R., and J. E. Lincoln. 1988. *Engineering Data Compendium: Human Perception and Performance*. AAMRL, WPAFB, Ohio, Vol. 1–3. New York: John Wiley and Sons.

Bosman, D. 1989. *Display Engineering: Conditioning, Technologies, Applications*. Amsterdam: Elsevier Science.

Brennesholtz, S. M., and E. H. Stupp. 2008. *Projection Displays*. New York: John Wiley & Sons.

Brodie, J., J. Chattratichart, M. Perry, and R. Scane. 2003. How age can inform the future design of the mobile phone experience. In *Universal Access in HCI: Inclusive Design in the Information Society*, ed. C. Stephanidis, 822–6. Mahwah, NJ: LEA.

Brown, S. F. 2008. hands-on computing: how multi-touch screens could change the way we interact with computers and each other. In *Scientific American Magazine*, July 2008.

Buchner, A., S. Mayr, and M. Brandt. 2009. The advantage of positive text-background polarity is due to high display luminance. *Ergonomics* 52(7):882–6.

Budin J.-P. 2003. Emissive displays: The relative merits of ACTFEL. In *Display Systems—Design and Applications*, 2nd ed., ed. L. MacDonald and A. Lowe, 191–219. New York: John Wiley & Sons.

Budinger T. F. 1984. An analysis of 3-D display strategies, Processing and Display of Three-Dimensional Data II. In *Proceedings of SPIE*, Vol. 507, 2–8.

Calero Valdez, A., M. Ziefle, A. Horstmann, D. Herding, and U. Schroeder. 2009. Effects of aging and domain knowledge on usability in small screen devices for diabetes patients. In *HCI and Usability for e-Inclusion*, LNCS 5889. ed. A. Holzinger and K. Miesenberger, 366–86. Berlin, Heidelberg: Springer.

Calero Valdez, A., M. Ziefle, U. Schroeder, A. Horstmann, and D. Herding. 2010. Task performance in mobile and ambient interfaces. Does size matter for usability of electronic diabetes assistants? In *Proceedings of the International Conference on Information Society (i-Soeciety 2010/IEEE)*, ed. C. A Shoniregun and G. A. Akmayeva, 526–33. London: Infonomics Society.

Çakir, A., D. J. Hart, and T. F. M. Stewart. 1979. *The VDT Manual—Ergonomics, Workplace Design, Health and Safety, Task Organization*. Darmstadt: IFRA (Inca-Fiej Research Association).

Castellano, J. A. 1992. *Handbook of Display Technology*. San Diego: Academic.

Castillo, M. 2010. E-Readers and E-Paper. *Am J Neuroradiol* 31:1–2.

Chen, J., W. Cranton, and M. Fihn. 2011. *Handbook of Visual Display Technology*. Berlin: Springer.

Chen M.-T. and C.-C. Lin. 2004. Comparison of TFT-LCD and CRT on visual recognition and subjective preference. Int J Ind Ergon 34:167–74.

Crawford, G. 2005. *Flexible Flat Panel Displays*. New York: John Wiley & Sons.

Cruz-Neira, C., D. J. Sandin, and A. T. DeFanti. 1993. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *Proceedings of SIGGRAPH '93*, 135–42. Anaheim, CA: ACM Press.

DeFanti, T. A., G. Dawe, D. J. Sandin, J. P. Schulze, P. O. J. Girado, F. Kuester, L. Smarr, and R. Rao. 2009. The StarCAVE, a third-generation CAVE and virtual reality OptIPortal. *Future Gener Comput Syst* 25(2):169–78.

Dillon, A. 1992. Reading from paper versus screens: A critical review of the empirical literature. *Ergonomics* 35(10):1297–326.

Downing, E., L. Hesselink, J. Ralston, and R. Macfarlane. 1996. A three-color, solid-state, three-dimensional display. *Science* 273(5279):1185–89.

Duncan, G.-R. 2008. Electronic paper targets colour video. Nat Photon 2(4):204–5.

Duncan, J., and G. Humphreys. 1989. Visual search and stimulus similarity. *Psychol Rev* 96(3):433–58.

Ellemberg, D., T. L. Lewis, C. H. Liu, and D. Maurer. 1999. Development of spatial and temporal vision during childhood. *Vision Res* 39:2325–33.

Fellowes, D. A., M. V. Wood, A. R. Hastings, D. S. Russell, A. K. Lum, A. P. Ghosh, O. Prache, and I. Wacyk. 2009. Active matrix organic light emitting diode (AMOLED)-XL performance and life test results. In *Proc. SPIE Vol. 73260F*. Display Concepts and Technologies. Orlando, FL.

Fruehauf, N., T. Aye, K. Yua, Y. Zou, and G. Savant. 2000. Liquid crystal digital scanner-based HMD. In *Proc. SPIE Helmet-and Head-Mounted Displays* Vol. 4021, 2–10. Orlando, FL.

Goldstein, M., G. Öqvist, M. Bayat, P. Ljungstrand, and S. Björk. 2001. Enhancing the reading experience: Using adaptive and sonified RSVP for reading on small displays. In *Proceedings of the Mobile HCI, 2001*, 1–9. Berlin: Springer.

Gould, J., L. Alfaro, V. Barnes, R. Finn, N. Grischkowsky, and A. Minuto. 1987. Reading is slower from CRT displays than from paper: Attempts to isolate a single-variable explanation. *Human Factors, 29*(3):269–99.

Graham-Rowe, D. 2007. Photonic fabrics take shape. *Nat Photon* 1(1):6–7.

Gröger, T., M. Ziefle, and D. Sommer. 2003. Anisotropic characteristics of LCD TFTs and their impact on visual performance. In *Human-Centred Computing: Cognitive, Social and Ergonomic Aspects*, ed. D. Harris, V. Duffy, M. Smith and C. Stephanidis, 33–7. Mahwah, NJ: LEA.

Heikenfeld, J. 2010. Lite, brite displays. In IEEE Spectrum International 3.10.

Heikenfeld, J., K. Zhou, E. Kreit, B. Raj, S. Yang, B. Sun, A. Milarcik, A. Clapp, and R. Schwartz. 2009. Electrofluidic displays using Young-Laplace transposition of brilliant pigment dispersions. *Nat Photonics* 3(5):292–6.

Helander, M., S. Little, and C. Drury. 2000. Adaptation and sensitivity to postural changes in sitting. *Hum Factors* 42(4):617–29.

Heppner, F., J. Anderson, A. Farstrup, and N. Weidenman. 1985. Reading performance on standardized test is better from print than from computer display. *J Read* 28:321–5.

Hilton, P. J. 2008. Ultra-wide FOV Retinal Display. In *Handheld Usability*, ed. S. Weiss. New York: John Wiley.

Hollands, J., H. Cassidy, S. McFadden, and R. Boothby. 2001. LCD versus CRT Displays: Visual search for colored symbols. In *Proceedings of the Human Factors and Ergonomics Society 45th Annual Meeting*, 1353–55. Santa Monica: Human Factors Society.

Hollands, J., H. Parker, S. McFadden, and R. Boothby. 2002. LCD versus CRT Displays: A Comparison of visual search performance for colored symbols. *Hum Factors* 44(2):210–21.

Holmes, R. 2003. Head-mounted display technology in virtual reality systems. In *Display Systems—Design and Applications*, 2nd ed., ed. L. MacDonald and A. Lowe, 61–82. New York: John Wiley & Sons.

Holzel, T. 1999. Are head-mounted displays going anywhere? *Inf Disp* 15(10):16–8.

Honda, T. 2000. Three-dimensional display technology satisfying “super multiview condition”. In *Proceedings Three-Dimensional Video and Display: Devices and Systems*, ed. B. Javidi and F. Okano, Vol. CR76, 218–49. San Jose, CA: SPIE Press.

Hornbeck, L. J. 2001. The DMD™ projection display chip: A MEMS-based technology. *MRS Bull* 26:325–7.

Howarth, P., and H. Istance. 1985. The association between visual discomfort and the use of visual display units. *Behav Inf Technol* 4:131–49.

Huang, D.-L., P. Rau, and Y. Liu. 2009. Effects of font size, display resolution and task type on reading Chinese fonts from mobile devices. *Int J Ind Ergon* 39:81–9.

Hung, G., K. Ciuffreda, and J. Semmlow. 1986. Static vergence and accomodation: Population norms and orthopic effects. *Doc Ophthalmol* 62:165–79.

Informa Telecoms and Media Global mobile forecasts. 2008. http://www.intomobile.com/2008/.

Iwasaki, T., and S. Kurimoto. 1988. Eye-strain and changes in accomodation of the eye and in visual evoked potential following quantified visual load. *Ergonomics* 31(12):1743–51.

Jaschinski, W., M. Bonacker, and E. Alshuth. 1996. Accommodation, convergence, pupil and eye blinks at a CRT-display flickering near fusion limit. *Ergonomics* 19:152–64.

Jhuo, L.-C., C.-W. Wu, and C.-C. Hu. 2009. A Resistive Multi-Touch Screen Integrated into LCD. In *SID Symposium Digest* 40:1187–88.

Jorke, H., and M. Fritz. 2006. Stereo projection using interference filters. *Stereoscopic Displays and Virtual Reality Systems XIII*. Stereoscopic Displays and Applications Proc. SPIE 6055. ed. A. J. Woods, N. A. Dodgson, J. O. Merritt, M. T. Bolas, and I. E. McDowall, 60550G.

King, C. N. 1996. Electroluminescent displays, SID 96 Seminar Lecture Notes, M-9, 1–36.

Kirsch, C. 2004. Laptops als Ersatz fur Desktop-PCs. Überall im Büro [Laptops as a replacement for desktop PCs. The office is everywhere]. iX, 12, 40–5.

Kleiber, M., and T. Alexander. 2011. Evaluation of a mobile AR telemaintenance system. In *Proceedings of the 14th International Conference on Human-Computer Interaction*. Orlando, FL. USA: Springer.

Kleiber, M., and C. Winkelholz. 2008. Distortion of depth perception in virtual environments using stereoscopic displays: quantitative assessment and corrective measures. In *Proceedings of the Stereoscopic Displays and Applications XIX conference, SPIE*, Vol. 6803. San Jose, CA: SPIE Press.

Kline, D. W., and C. T. Scialfa. 1997. Sensory and perceptual functioning: Basic research and human factors implications. In *Handbook of Human Factors and the Older Adult*, ed. A. Fisk and W. Rogers, 27–54. San Diego: Academic.

Kokoschka, S., and P. Haubner. 1986. Luminance ratios at visual display workstations and visual performance. *Lighting Res Technol* 17(3):138–44.

Kothiyal, K., and S. Tettey. 2001. Anthropometry for design for the elderly. *Int J Occup Saf Ergon* 7(1):15–34.

Kruger, W., C. A. Bonh, B. Frohlich, H. Schuth, W. Strauss, and G. Wesche. 1995. The Responsive Workbench: A virtual work environment. *Computer* 28(7):42–8.

Kubota, S. 1997. Effects of reflection properties of liquid-crystal displays on subjective ratings of disturbing reflected glare. *J Light Vis Environ* 21:33–42.

Lambooij, M. T. M., W. A. IJsselsteijn, and I. Heynderickx. 2007. Visual discomfort in stereoscopic displays: A review, in Stereoscopic Displays and Virtual Reality Systems XIV. In *Proceedings of the SPIE*, Vol. 6490. San Jose, CA: SPIE Press.

Langhans, K., D. Bezecny, D. Homann, D. Bahr, C. Vogt, C. Blohm, and K.-H. Scharschmidt. 1998. New Portable FELIX 3D Display. In *Proc SPIE, vol. 3296, SPIE—Int’l Soc for Optical Eng*., 204–16. San Jose, CA: SPIE Press.

Leister, N., A. Schwerdtner, G. Fütterer, S. Buschbeck, J.-C. Olaya, and S. Flon. 2008. Full-color interactive holographic projection system for large 3D scene reconstruction. In *Proc SPIE*, Vol. 6911. San Jose, CA: SPIE Press.

Li, Z., A. K. Bhowmik, and P. J. Bos. 2008. Introduction to mobile displays. In *Mobile Displays*, ed. P. Bhowmik, Z. Li, and P. Bos, 1–22. Essex, England: John Wiley & Sons Ltd.

Lin, Y.-T., P.-H. Lin, S.-L. Hwang, S.-C. Jeng, and C.-C. Liao. 2009. Investigation of legibility and visual fatigue for simulated flexible electronic paper under various surface treatments and ambient illumination conditions. *Appl Ergon* 40:922–8.

Lincoln, J. 2010. March of the pico projectors. In *IEEE Spectrum: Inside Technology 5.10*.

Lippert, T. M. 2006. Display Devices: RSD (Retinal Scanning Display). In *Avionics: Elements, Software and Functions, The Avionics Handbook*, ed. C. R. Spitzer. Boca Raton: CRC Press.

Luczak, H., and O. Oehme. 2002. Visual Displays—developments of the past, the present and the future. In *Proceedings of the 6th International Scientific Conference on Work with Display Units*, ed. H. Luczak, A. Çakir, and G. Çakir, 2–5. Berlin: Ergonomic Institute.

Luczak, H., M. Park, B. Balazs, S. Wiedenmaier, and L. Schmidt. 2003. Task performance with a wearable augmented reality interface for welding. In *Human-Computer Interaction. Cognitive, Social and Ergonomic Aspects*, ed. D. Harris, V. Duffy, M. Smith, and C. Stephanidis, 98–102. Mahwah, NJ: LEA.

Lüder, E. 2003. Active matrix addressing of LCDs: Merits and shortcomings. In *Display Systems—Design and Applications*, 2nd ed., ed. L. MacDonald and A. Lowe, 157–71. New York: John Wiley & Sons.

Lueder, E. 2010. *Liquid Crystal Displays—Addressing Schemes and Electro-Optical Effect*. Chichester: Wiley Series in Display Technology.

Macadam, D. L. 1982. *Caliometry. American Institute of Physics Handbook* 6:182–97. New York: McGraw-Hill.

Maxwell, I. 2007. An Overview of Optical-Touch Technologies. In *Information Display* 12/07.

Menozzi, M., F. Lang, U. Näpflin, C. Zeller, and H. Krueger. 2001. CRT versus LCD: Effects of refresh rate, display technology and background luminance in visual performance. *Displays* 22(3):79–85.

Mertens, A., N. Jochems, C. M. Schlick, D. Dünnebacke, and J. H. Dornberg. 2010. A novel input method for trepidant users of telemedical services. In *Advances in Human Factors and Ergonomics in Healthcare*, ed. V. Duffy, 662–71. USA: CRC Press.

Mertens, A., B. Kausch, D. Dünnebacke, P. Laing. 2009. Adequate Requirements Analysis in Homely Rehab. In *eChallenges e-2009 Conference Proceedings*, ed. P. Cunningham and M. Cunningham. Istanbul, Turkey: IIMC International Information Management Corporation.

Mertens, A., C. Wacharamanotham, J. Hurtmanns, M. Kronenbuerger, P. H. Kraus, A. Hoffmann, C. Schlick, and J. Borchers. 2011. Model-based processing of swabbing movements on touch screens to improve accuracy and efficacy for information input of individuals suffering from kinetic tremor. In *Human-Computer Systems Interaction. Backgrounds and Applications 2, Advances in Soft Computing*. Berlin: Springer.

Miles, M. W. 2004. *Interferometric Modulation: MOEMS as an Enabling Technology for High-Performance Reflective Displays*, 131. San Francisco: Iridigm Display Corp. Proc. SPIE 4985.

Miyao, M., S. Hacisalihzade, J. Allen, and L. Stark. 1989. Effects of VDT resolution on visual fatigue and readability: An eye movement approach. *Ergonomics* 32(6):603–14.

Mphepo, W., Y.-P. Huang, P. Rudquist, and H.-P. D. Shieh. 2010. Digital micro hinge (DMH) based display pixels. *J Disp Technol* 6(4):142–9.

Myeon-Cheon, C., K. Youngkyoo, and H. Chang-Sik. 2008. Polymers for flexible displays: From material selection to device applications. *Prog Polym Sci* 33(6):581–630.

Nelson, T. J., and J. R. Wullert II. 1997. *Electronic Information Display Technologies* (Series on information display, Vol. 3). River Edge, NJ: World Scientific.

Norman, J. F., J. T. Todd, V. J. Perotti, and J. S. Tittle. 1996. The visual perception of three-dimensional length. *J Exp Psychol Hum Percept Perform* 22(1):173–86.

Odenthal, B., M. Mayer, W. Kabuß, B. Kausch, and C. Schlick. 2009. Investigation of error detection in assembled workpieces using an augmented vision system, In *Proceedings of the IEA2009—17th World Congress on Ergonomics*, 1–9. Beijing, China.

Oehme, O., L. Schmidt, and H. Luczak. 2003. Comparison between the strain indicator HRV of a head-based virtual retinal display and LC head mounted displays for augmented reality. *Int J Occup Saf Ergon* 9(4):411–22.

Oehme, O., S. Wiedenmaier, L. Schmidt, and H. Luczak. 2001. Empirical studies on an augmented reality user interface for a head based virtual retinal display. In *Usability Evaluation and Interface Design: Cognitive Engineering, Intelligent Agents and Virtual Reality*, ed. M. Smith, G. Salvendy, D. Harris, and R. Koubek, 1026–30. Mahwah, NJ: LEA.

Oetjen, S., and M. Ziefle. 2004. Effects of anisotropy on visual performance regarding different font sizes. In *Work with Computing Systems*, ed. H. Khalid, M. Helander, and A. Yeo, 442–7. Kuala Lumpur, Malaysia: Damai Sciences.

Oetjen, S., and M. Ziefle. 2007a. The effects of LCD anisotropy on the visual performance of users of different ages. *Hum Factors* 49(4):619–27.

Oetjen, S., and M. Ziefle. 2007b. Children working with computers: The effects of user's age and task complexity. In *Work with Computing Systems*, ed. A. Toomingas, A. Lantz, and Th. Berns. Stockholm: Royal Institute of Technology.

Oetjen, S., and M. Ziefle. 2009. A visual ergonomic evaluation of different screen technologies. *Appl Ergon* 40:69–81.

Oetjen, S., M. Ziefle, and T. Gröger. 2005. Work with visually suboptimal displays—in what ways is the visual performance influenced when CRT and TFT displays are compared? In *Proceedings of the HCI International 2005. Vol. 4: Theories, Models and Processes in Human Computer Interaction*. St. Louis, MO: Mira Digital Publisher.

Omori, M., T. Watanabe, J. Takai, H. Takada, and M. Miyao. 2002. Visibility and characteristics of the mobile phones for elderly people. *Behav Inf Technol* 21(5):313–6.

Ono, Y. A. 1993. *Electroluminescent Displays, Seminar Lecture Notes, F-1/1-30*. Santa Ana, CA: Society for Information Display.

Owens, D., and K. Wolf-Kelly. 1987. Near work, visual fatigue, and variations of oculomotor tonus. *Invest Ophthalmol Vis Sci* 28(4):743–9.

Ozawa, L. 2007. *Cathodoluminescence and Photoluminescence: Theories and Practical Applications*. Boca Raton, FL: CRC Press, Taylor & Francis Group.

Park, M., L. Schmidt, and H. Luczak. 2005. Changes in hand-eye-coordination with different levels of camera displacement from natural eye position. In *10th International Conference on Human Aspects of Advanced Manufacturing: Agility and Hybrid Automation—HAAMAHA 2005*, 191–9. San Diego, CA.

Park, M., S. Serefoglou, L. Schmidt, K. Radermacher, C. Schlick, and H. Luczak. 2008. Hand-eye coordination using a video see-through augmented reality system. *Ergonomics Open J* 39–47.

Patterson, R., L. Moe, and T. Hewitt. 1992 Factors that affect depth perception in stereoscopic displays. *Hum Factors* 34:655–67.

Pfendler, C., and C. Schlick. 2007. A comparative study of mobile map displays in a geographic orientation task. *Behav Inf Technol* 26(2):455–63.

Pfendler, C., J. Thun, T. Alexander, C. Schlick. 2011. The influence of different electronic maps and displays on performance and operator state in geographic orientation task. In *Behaviour & Information Technology*, 1–12. Hampshire, England: Taylor & Francis.

Pfendler, C., H. Widdel, and C. Schlick. 2005. Bewertung eines Head-Mounted und eines Hand-Held Displays bei einer Zielerkennungsaufgabe [Task related evaluation of head-mounted and hand-held displays]. *Zeitschrift für Arbeitswissenschaft* 59(1):13–21.

Piccoli, B., M. D'Orso, P. L. Zambelli, P. Troiano, and R. Assini. 2001. Observation distance and blinking rate measurement during on-site investigation: New electronic equipment. *Ergonomics* 44(6):668–76.

Piccoli, B., G. Soci, P. L. Zambelli, and D. Pisaniello. 2004. Photometry in the workplace: the rationale for a new method. *Annals of Occupational Hygiene,* 48:29–38.

Pioneer. 2001. Why choose Plasma. http://www.pioneerelectronics.com/Pioneer/CDA/Common/ArticleDetails/0,1484,1547,00.html (accessed September 15, 2005).

Plainis, S., and I. Murray. 2000. Neurophysiological interpretation of human visual reaction times: Effect of contrast, spatial frequency and luminance. *Neuropsychologia* 38:1555–64.

Precht, M., N. Meier, and J. Kleinlein. 1997. *EDV-Grundwissen: Eine Einführung in Theorie und Praxis der Modernen EDV [Computer Basics: An Introduction in Theory and Application of Modern Computing]*. Bonn: Addison-Wesley-Longman.

Rahman, T., and P. Muter. 1999. Designing an interface to optimize reading with small display windows. *Hum Factors* 41(1):106–17.

Rao, B., and L. Minakakis. 2003. Evolution of Mobile Location-based Services. *Commun ACM* 46(12):61–5.

Saffer, D. 2008. *Designing Gestural Interfaces: Touchscreens and Interactive Devices*. Sebastopol, CA: O'Reilly Media.

Sanders, M. S., and E. J. McCormick. 1993. *Human Factors in Engineering and Design*. 7th ed. New York: McGraw-Hill.

Schadt, M. 1996. Optisch strukturierte Flüssigkeitskristall-Anzeigen mit großem Blickwinkelbereich [Optically structured liquid crystal displays with wide viewing angles]. *Physikalische Blätter* 52(7–8):695–8.

Schenkmann, B., T. Fukunda, and B. Persson. 1999. Glare from monitors measured with subjective scales and eye movements. *Displays* 20:11–21.

Schlick, C., R. Daude, H. Luczak, M. Weck, and J. Springer. 1997. Head-mounted display for supervisory control in autonomous production cells. *Displays* 17(3–4):199–206.

Schlick, C., B. Odenthal, M. Mayer, J. Neuhöfer, M. Grandt, B. Kausch, and S. Mütze-Niewöhner. 2009. Design and evaluation of an augmented vision system for self-optimizing assembly cells. In *Industrial Engineering and Ergonomics—Visions, Concepts, Methods and Tools—Festschrift in Honor of Professor Holger Luczak*, 539–60. Berlin: Springer, Hrsg.: Schlick, C.

Schlick, C., R. Reuth, and H. Luczak. 2000. Virtual Reality User Interface for Autonomous Production. In *Advances in Networked Enterprises*, ed. L. M. Camarinha-Matos, H. Afsarmanesh, and H. Erbe, 279–86. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Schlick, C., C. Winkelholz, F. Motz, and H. Luczak. 2006. Self-Generated Complexity and Human-Machine Interaction. *IEEE Trans Syst Man Cybern A Syst Hum* 36(1):220–32.

Schmidt, U. 2009. *Professionelle Videotechnik*. Berlin: Springer.

Schneider, N., S. Schreiber, J. Wilkes, M. Grandt, and C. Schlick. 2007. Investigation of adaptation dimensions for age-differentiated human-computer interfaces. In *Universal Access in HCI, Part I, HCII 2007, 12th International Conference on Human-Computer Interaction, Beijing, China*, ed. C. Stephanidis. Berlin: Springer.

Schowengerdt, B. T., and E. J. Seibel. 2006. True 3-D scanned voxel displays using single or multiple light sources. *J Soc Inf Disp* 14(135):135–43.

Sheedy, E. J. 2005. Office lighting for computer use. In *Visual Ergonomics Handbook*, ed. J. Anshel, 37–51. Boca Raton: CRC.

Sheedy, J., M. Subbaram, and J. Hayes. 2003. Filters on computer displays—effects on legibility, performance and comfort. *Behav Inf Technol* 22(6):427–33.

Shen, I.-H., K. K. Shieh, C.-Y. Chao, and D.-S. Lee. 2009. Lighting, font style, and polarity on visual performance and visual fatigue with electronic paper displays. *Displays* 30:53–8.

Shieh, K.-K., and C.-C. Lin. 2000. Effects of screen type, ambient illumination, and color combination on VDT visual performance and subjective preference. *Int J Ind Ergon* 26(5):527–36.

Sommerich, C., S. Joines, and J. Psihoios. 2001. Effects of computer monitor viewing angle and related factors on strain, performance, and preference outcomes. *Hum Factors* 43(1):39–55.

Stanney, K. M., and M. Zyda. 2002. Virtual environments in the 21st century. In *Handbook of Virtual Environments*, ed. K. M. Stanney, 1–14. Mahwah, NJ: LEA.

Stolle, H., J.-C. Olaya, S. Buschbeck, H. Sahm, and A. Schwerdtner. 2008. Technical solutions for a full-resolution autostereoscopic 2D/3D display technology. In *Proceedings SPIE, Volume 6803, Autostereoscopic Displays II*, ed. N. S. Holliman, and J. O. Merri. San Jose, CA.

Stone, P., A. Clarke, and A. Slater. 1980. The effect of task contrast on visual performance and visual fatigue at a constant illuminance. *Lighting Res Technol* 12:144–59.

Sun, B., and J. Heikenfeld. 2008. Observation and optical implications of oil dewetting patterns in electrowetting displays. *J Micromech Microeng* 18(2):1–8.

Takaki, Y., and N. Nago. 2010. Multi-projection of lenticular displays to construct a 256-view super multi-view display. *Opt Express* 18(9):8824.

Tech Crunchies—Internet Statistics and Numbers. http://techcrunchies.com/distribution-of-gsm-connections-worldwide/2009 (accessed January 15, 2010).

Theis, D. 1999. Display technologie. In *Vom Arbeitsplatzrechner zum ubiquitären Computer [From the Desktop PC to the ubiquitous Computer]*, ed. C. Müller-Schloer and B. Schallenberger, 205–38. Berlin: VDE.

Tumler, J., F. Doil, R. Mecke, G. Paul, M. Schenk, E. A. Pfister, A. Huckauf, L. Bockelmann, A. Roggentin. 2008. Mobile Augmented Reality in industrial applications: Approaches for solution of user-related issues. In *ISMAR '08: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. Washington, DC, USA: IEEE Computer Society.

Urey, H., D. W. Wine, and J. R. Lewis. 1999. Scanner design and resolution tradeoffs for miniature scanning displays. In *Proceedings SPIE, Flat Panel Display Technology and Display Metrology*, ed. B. Gnade and E. F. Kelley, Vol. 3636, 60–8. San Jose, CA: SPIE Press.

Van Schaik, P., and J. Ling. 2001. The effects of frame layout and differential background contrast on visual search performance in web pages. *Interact Comput* 13:513–25.

Vetter, S., N. Jochems, B. Kausch, S. Mütze-Niewöhner, and C. M. Schlick. 2010. Age-induced change in visual acuity and its impact on performance in a target detection task with electronic information displays. *Occup Ergon* 9(2):99–110.

Vogel, U., D. Kreye, B. Richter, and G. Bunk. 2008. OLED microdisplays: Advanced functionality and systems. In *SID-MID-Europe Chapter 2008*. Jena.

Von Waldkirch, M. 2004. "Retinal Projection Displays for Accommodation-Insensitive Viewing". Doctoral thesis, Zurich, Swiss Federal Institute of Technology, Aachen: Shaker.

Wang A.-H., and M. T. Chen. 2000. Effects of polarity and luminance contrast on visual performance and VDT display quality. *Int J Ind Ergon* 25:415–21.

Wang, A.-H., H.-T. Kui, and S.-C. Jeng. 2009. Effects of ambient illuminance on users' visual performance using various electronic displays. *J Soc Inf Disp* 17(8):665–9.

Weiss, S. 2002. Handheld Usability. New York: John Wiley.

Wiley, G. A., B. Steele, S. Saeed, and G. Raskin. 2008. Mobile Display Digital Interface (MDDI). In *Mobile Displays*, ed. A. Bhowmik, Z. Li, and P. Bos. Chichester, England: John Wiley & Sons Ltd.

Wilkins, A., I. Nimmo-Smith, A. Tait, C. McManus, S. Della Sala, A. Tilley, K. Arnold, M. Barrie, and S. Scott. 1984. A neurological basis for visual discomfort. *Brain* 107:989–1017.

Winkelholz, C. 2008. Theoretische Untersuchung zur Schärfentiefe eines Retinal Laser Scanning Displays. In *Ergonomie und Mensch-Maschine-Systeme*, ed. L. Schmidt, C. Schlick, and J. Grosche, 405–22. Berlin: Springer Verlag.

Winkelholz, C., M. Kleiber, and C. Schlick. 2010a. Analysis of the Variability of Three-Dimensional Spatial Relations in Visual Short-Term Memory. In *Proceedings of the 32nd Annual Conference of the Cognitive Science Society, August 11–14, 2010 in Portland, Oregon, USA*, ed. S. Ohlsson and R. Catrambone, 1679–94. Austin, TX: Cognitive Science Society.

Winkelholz, C., M. Kleiber, and C. Schlick. 2010b. Modeling the cognitive representation of basic three-dimensional spatial relations in visual short-term memory, In *IEEE International Conference on Systems, Man, and Cybernetics, 10–13 October 2010*, 1838–45. Istanbul, Turkey, Istanbul 2010.

Wolf, E., and A. Schraffa. 1964. Relationship between critical flicker frequency and age in flicker perimetry. *Arch Ophthalmol* 72: 832–43.

Woodson, W. E. 1987. *Human Factor Reference Guide for Electronics and Computer Professionals*. New York: McGraw-Hill.

Yagi, I., N. Hirai, Y. Miyamoto, M. Noda, A. Imaoka, N. Yoneya, K. Nomoto, J. Kasahara, A. Yumoto, and T. Urabe. 2008. A flexible full-color AMOLED display driven by OTFTs. *J Soc Inf Disp* 16(15):15–20.

Yeh, P. 2009. *Optics of Liquid Cristal Displays*. New York: John Wiley & Sons.

Yeh, Y.-Y., and C. Wickens. 1984. Why do performance and subjective workload measures dissociate? In *Proceedings of the 28th Annual meeting of the Human Factors Society*, 504–8. Santa Monica: Human Factors and Ergonomics Society.

Zehner, R. 2008. Electronic paper displays. In *Mobile Displays*, ed. A. Bhowmik, Z. Li, and P. Bos. Chichester, West Sussex: Wiley Series in Display Technology.

Ziefle, M. 1998. Effects of display resolution on visual performance. *Hum Factors* 40(4):554–68.

Ziefle, M. 2001a. CRT screens or TFT displays? A detailed analysis of TFT screens for reading efficiency. In *Usability Evaluation and Interface Design*, ed. M. Smith, G. Salvendy, D. Harris, and R. Koubek, 549–53. Mahwah: LEA.

Ziefle, M. 2001b. Aging, visual performance and eyestrain in different screen technologies. In *Proceedings of the Human Factors and Ergonomics Society 45th Annual Meeting*, 262–6. Santa Monica: Human Factors Society.

Ziefle, M. 2002. *Lesen am Bildschirm [Reading from Screens]*. Münster, Germany: Waxmann.

Ziefle, M. 2003a. Users with body heights above and below the average: How adequate is the standard VDU setting with respect to visual performance and muscular load? In *Human Factors in Organizational Design and Management*, ed. H. Luczak and K. J. Zink, 489–94. Santa Monica: IEA Press.

Ziefle, M. 2003b. Sitting posture, postural discomfort, and visual performance: A critical view on the independence of cognitive and anthropometric factors in the VDU workplace. *Int J Occup Saf Ergon* 9(4):495–506.

Ziefle, M. 2008. Instruction format and navigation aids in mobile devices. In *Usability and Human Computer Interaction for Education and Work*, ed. A. Holzinger, 339–58. Berlin: Springer.

Ziefle, M. 2009. Visual ergonomic issues in LCD-Displays. An insight into working conditions and user characteristics. In *Methods and Tools of Industrial Engineering and Ergonomics for Engineering Design, production, and Service-Traditions, Trends and Vision*, ed. C. M. Schlick, 561–72. Berlin: Springer.

Ziefle, M. 2010a. Information presentation in small screen devices: The trade-off between visual density and menu foresight. *Applied Ergonomics* 41(6):719–30.

Ziefle, M. 2010b. Modelling mobile devices for the elderly. In *Advances in Ergonomics Modeling and Usability Evaluation*, ed. H. Khalid, A. Hedge, and T. Z. Ahram, 280–90. Boca Raton: CRC Press.

Ziefle, M., and S. Bay. 2005. How older adults meet cognitive complexity: Aging effects on the usability of different cellular phones. *Behav Inf Technol* 24(5):375–89.

Ziefle, M., and S. Bay. 2006. How to overcome disorientation in mobile phone menus: a comparison of two different types of navigation aids. *Hum Comput Interact* 21(4):393–433.

Ziefle, M., and S. Bay. 2008. Transgenerational Designs in Mobile Technology. In *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, ed. J. Lumsden, 122–40. Hershey, PA: IGI Global.

Ziefle, M., T. Gröger, and D. Sommer. 2003. Visual costs of the inhomogeneity of contrast and luminance by viewing TFT-LCD screens off-axis. *Int J Occup Saf Ergon* 9(4):507–17.

Ziefle, M., O. Oehme, and H. Luczak. 2005. Information presentation and visual performance in head-mounted displays with augmented reality. *Zeitschrift für Arbeitswissenschaft* 59(3–4):331–44.

Zingale, C., V. Ahlstrom, and B. Kudrick. 2005. *Human Factors Guidance for the Use of Handheld, Portable, and Wearable Computing Devices*. Technical Report, DOT/FAA/CT-05/15. Springfield, Virginia: National Technical Information Service.

9 Haptic Interface

Hiroo Iwata

CONTENTS

9.1 INTRODUCTION

It is well known that sense of touch is inevitable for understanding the real world. The use of force feedback to enhance computer–human interaction (HCI) has often been discussed. A haptic interface is a feedback device that generates sensation to the skin and muscles, including a sense of touch, weight, and rigidity. Compared with ordinary visual and auditory sensations, haptics is difficult to synthesize. Visual and auditory sensations are gathered by specialized organs, the eyes and ears. On the other hand, a sensation of force can occur at any part of the human body and is therefore inseparable from actual physical contact. These characteristics lead to many difficulties when developing a haptic interface. Visual and auditory media are widely used in everyday life, although little application of haptic interfaces is used for information media.

In the field of virtual reality, haptic interface is one of the major research areas. The last decade has seen significant advances in the development of haptic interfaces. High-performance haptic devices have been developed and some of them are commercially available. This chapter presents current methods and issues in developing haptic interfaces.

Section 9.2 describes the mechanism of haptic sensation and overall view of feedback technologies. This section is followed by three sections (9.3, 9.5, 9.6, and 9.9) that introduce examples of haptic interface technologies developed by the author. Section 9.7 presents application areas and future prospects of haptic interfaces.

9.2 MECHANISM OF HAPTICS AND METHODS FOR HAPTIC FEEDBACK

9.2.1 Somatic Sensation

Haptic interface presents synthetic stimulation to somatic sensation. Somatic sensation comprises proprioception and skin sensation. Proprioception is complemented by mechanoreceptors of skeletal articulations and muscles. There are three types of joint position receptors: (1) free nerve endings, (2) Ruffini corpuscles, and (3) Pacinian corpuscles. The Ruffini corpuscle detects static force. On the other hand, Pacinian corpuscle has a function to measure acceleration of the joint angle. Position and motion of the human body is perceived by these receptors. Force sensation is derived from mechanoreceptors of muscles, muscle spindles, and golgi tendons. These receptors detect contact forces applied by an obstacle in the environment.

Skin sensation is derived from mechanoreceptors and thermoreceptors of the skin. The sense of touch is evoked by these receptors. Mechanoreceptors of the skin are classified into four types: (1) Merkel disks, (2) Ruffini capsules, (3) Meissner corpuscles, and (4) Pacinian corpuscles. These receptors detect edge of object, skin stretch, velocity, and vibration, respectively.

9.2.2 Proprioception and Force Display

Force display is a mechanical device that generates a reaction force from virtual objects. Haptic interfaces have recently become a rapidly growing research area, although the technology is still in a state of trial and error. There are several approaches to implementing haptic interfaces, which are described in Sections 9.2.2.1 through 9.2.2.4.

9.2.2.1 Exoskeleton-Type Force Display

An exoskeleton is a set of actuators attached to a hand or a body. In the field of robotics research, exoskeletons have often been used as master manipulators for teleoperations. However, most master manipulators entail a large amount of hardware and therefore have a high cost associated with them, which restricts their application areas. Compact hardware is needed in order to use them in human–computer interactions. The first example of a compact exoskeleton suitable for desktop use was published in 1990 (Iwata 1990a,b). The device applies force to the fingertips as well as the palm. Figure 9.1 shows the overall view of the system.

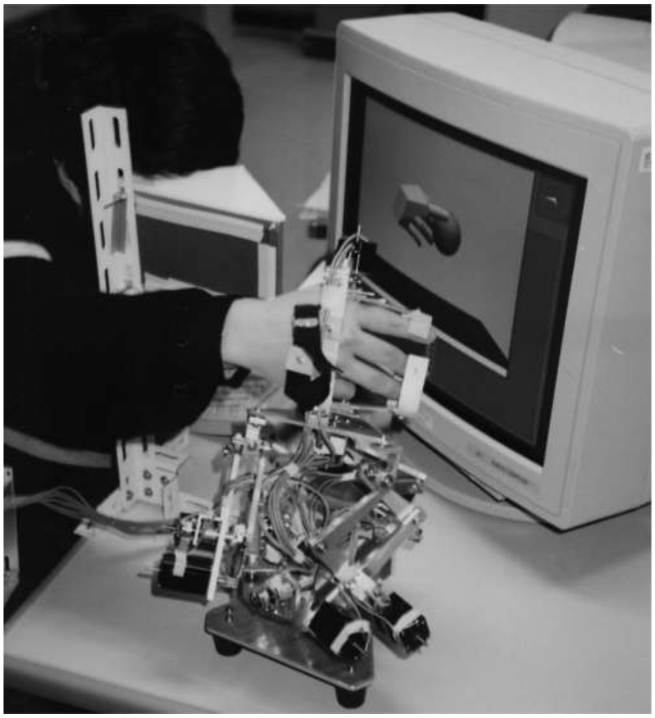


FIGURE 9.1 Overall view of a desktop force display system.

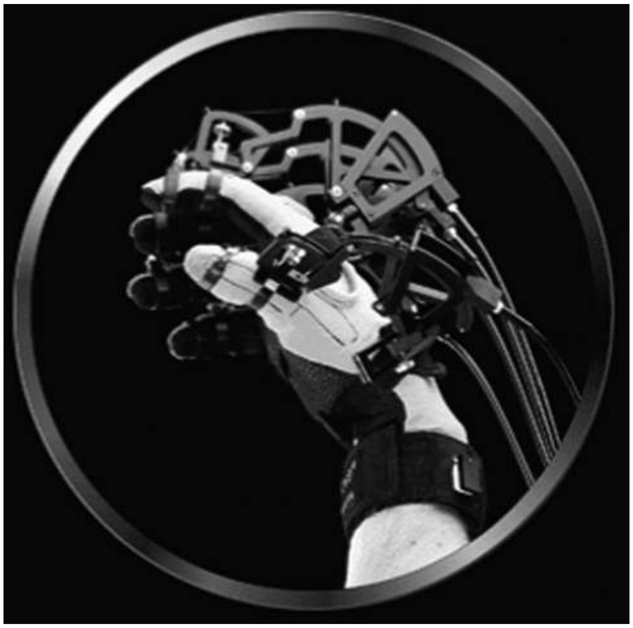


FIGURE 9.2 CyberGrasp, a commercially available exoskeleton in which cables are used to transmit force.

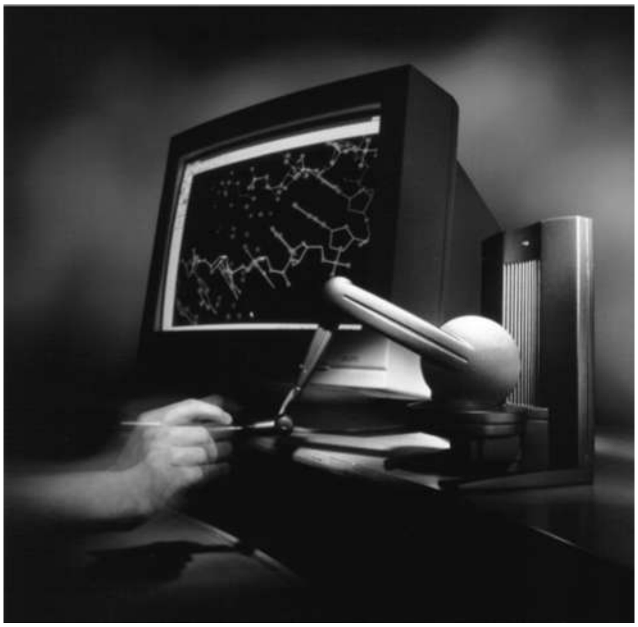


FIGURE 9.3 PHANToM, one of the most popular commercially available haptic interfaces.

Lightweight and portable exoskeletons have also been developed. Burdea et al. (1992) used small pneumatic cylinders to apply force to the fingertips. CyberGrasp (Figure 9.2) is a commercially available exoskeleton in which cables are used to transmit force (http://www.vti.com).

9.2.2.2 Tool-Handling-Type Force Display

A tool-handling-type force display is the easiest way to realize force feedback. The configuration of this type of display is similar to that of a joystick. Unlike an exoskeleton, the tool-handling-type force display is free from the need to be fitted to a user's hand. It cannot generate a force between the fingers, but it has practical advantages.

A typical example of this category is the pen-based force display (Iwata 1993). A pen-shaped grip is supported by two pantographs of three degrees of freedom (DOFs) that enables a six-DOF force/torque feedback. Another example of this type of display is the HapticMaster, which was demonstrated at the Edge venue of SIGGRAPH '94. The device has a ball-shaped grip to which six-DOF force/torque is fed back (Iwata 1994). This device employs a parallel mechanism in which a top triangular platform and a base triangular platform are connected by three sets of pantographs. This compact hardware has the ability to carry a large payload.

Massie and Salisbury (1994) developed the PHANToM, which has a three-DOF pantograph. A thimble with a gimbal is connected to the end of the pantograph, which can then apply a three-DOF force to the fingertips. The PHANToM has become one of the most popular commercially available haptic interfaces (Figure 9.3).

9.2.2.3 Object-Oriented-Type Force Display

The object-oriented-type force display is a radical idea for the design of a haptic interface. The device moves or deforms to simulate the shapes of virtual objects. A user of the device can come into physical contact with the virtual object through its surface.

An example of this type of display can be found in the work by Tachi et al. (1994). Their device consists of a shape approximation prop mounted on a manipulator. The position of the fingertip is measured and the prop moves to provide a contact point for the virtual object. McNeely (1993) proposed an idea "robotic graphics," which is similar to Tachi's method. Hirose (Hirota and Hirose 1996) developed a surface display that creates a contact surface using a 4 × 4 linear actuator array. The device simulates an edge or a vertex of a virtual object.

9.2.2.4 Passive Prop

A passive input device equipped with force sensors is a different approach to developing a haptic interface. Murakami and Nakajima (1994) used a flexible prop to manipulate a three-dimensional (3D) virtual object. The force applied by a user is measured and the deformation of the virtual object is determined based on the applied force. Sinclair (1997) developed a force sensor array to measure pressure distribution. These passive devices allow users to interact using their bare fingers. However, these devices have no actuators, so they cannot represent the shape of virtual objects.

9.2.3 Proprioception and Full-Body Haptics

One of the new frontiers of haptic interface development is full-body haptics that includes foot haptics. Forces applied to a whole body play very important roles in locomotion. The most intuitive way to move about the real world is to walk on foot. Locomotion interface is a device that provides the sense of walking while the walker's body is maintained localized in the real world. There are several approaches to realize locomotion interfaces, which are discussed in Sections 9.2.3.1 through 9.2.3.5.

9.2.3.1 Sliding Device

A project named Virtual Perambulator was aimed at developing locomotion interfaces using a specialized sliding device (Iwata and Fujii 1996). The primary object of the first stage was to allow for the changing direction of a walker's feet. Controlling steering bars or joysticks is not as intuitive in this case as in locomotion. The first prototype of the Virtual Perambulator was developed in 1989 (Iwata and Matsuda 1992). Figure 9.4 shows the overall view of the apparatus. A user of the system wears a parachute-like harness and omnidirectional roller skates. The trunk of the walker is fixed to the framework of the system by a harness. An omnidirectional sliding device is used for changing the direction using the feet. A specialized roller skate equipped with four casters was developed, which enabled two-dimensional (2D) motion. The walker could freely move his or her feet in any direction. Motion of the feet was measured by an ultrasonic range detector. From the result of this measurement, an image of the virtual space was displayed in the head-mounted display corresponding with the motion of the walker. The direction of locomotion in virtual space was determined according to the direction of the walker's step.

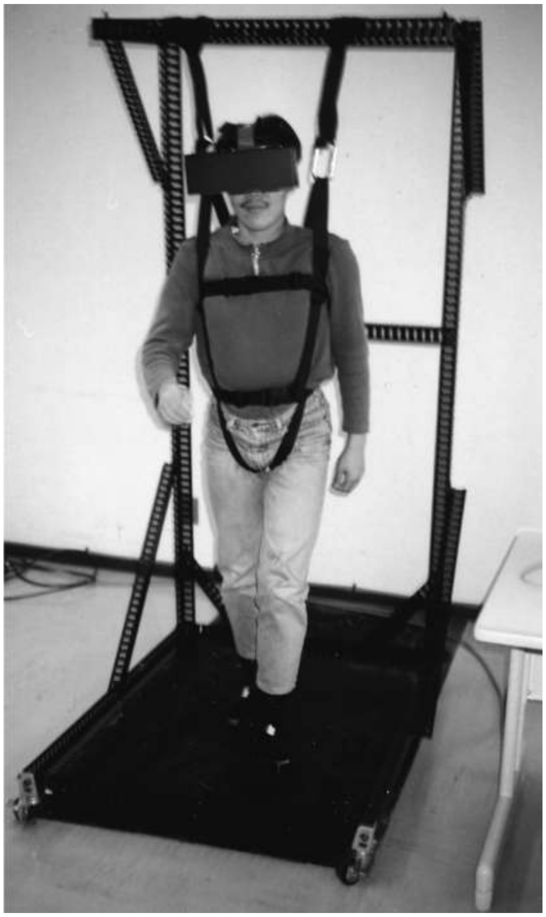


FIGURE 9.4 The first prototype of Virtual Perambulator.

9.2.3.2 Treadmill

A simple device for virtual walking is a treadmill, ordinarily used for maintaining physical fitness. An application of this device to virtual building simulation was developed at the University of North Carolina (UNC) (Brooks 1986). The treadmill has a steering bar similar to that of a bicycle. A treadmill equipped with a series of linear actuators underneath the belt was developed at Advanced Telecommunication Research (ATR) (Noma, Sugihara, and Miyasato 2000). The device is named GSS, which simulates the slope of virtual terrain. The TreadPort developed at the University of Utah is a treadmill that is combined with a large manipulator connected to a walker (Christensen et al. 1998). The manipulator provides gravitational force while the walker is passing a slope. Figure 9.5 shows the overall view of a TreadPort.

The omnidirectional treadmill employs two perpendicular treadmills, one inside the other. Each belt is made from approximately 3400 separate rollers, woven together into a mechanical fabric. Motion of the lower belt is transmitted by the rollers to a walker. This mechanism enables omnidirectional walking (Darken, Cockayne, and Carmein 1997).

9.2.3.3 Footpad

Footpad applied to each foot is an alternative implementation of a locomotion interface. Two large manipulators driven by hydraulic actuators were developed at the University of Utah and applied to a locomotion interface. These manipulators are attached to the feet of a walker. The device is named BiPort (http://www.sarcos.com). These manipulators can present the viscosity of virtual ground. A similar device has been developed at the Cybernet Systems Corporation, Michigan, which uses two three-DOF motion platforms for the feet (Poston et al. 1997). These devices, however, have not been evaluated or applied to virtual environments.

9.2.3.4 Pedaling Device

In the battlefield simulator of the NPSNET project, a unicycle-like pedaling device was used for locomotion in a

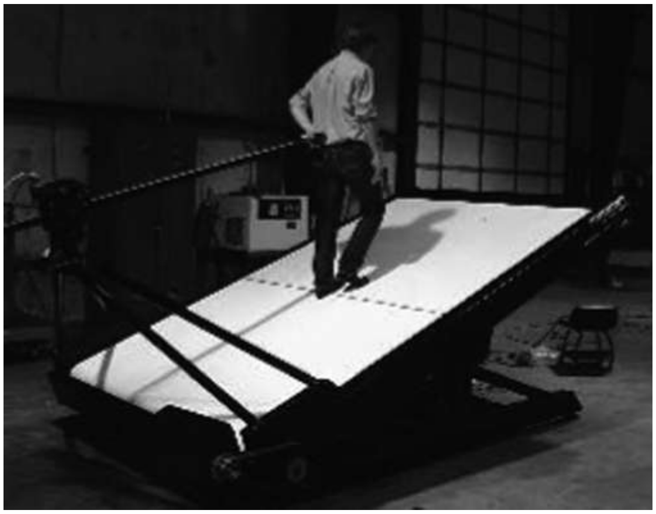


FIGURE 9.5 Overall view of a TreadPort.

virtual battlefield (Prat et al. 1994). A player of the system changes direction by twisting his or her waist.

The OSIRIS, a simulator of night-vision battle, utilizes a stair stepper device (Lorenzo et al. 1995). A player of the system changes direction by controlling the joystick or twisting his or her waist.

9.2.3.5 Gesture Recognition of Walking

Slater et al. (1994) proposed locomotion in virtual environments by "walking in place." They recognized the gesture of walking using a position sensor and a neural network.

9.2.4 Skin Sensation and Tactile Display

The tactile display that stimulates skin sensation is a well-known technology. It has been applied to communication aids for blind individuals as well as manipulator. A sense of vibration is relatively easy to produce, and a good deal of work has been done using vibration displays (Kontarinis and Howe 1995; Minsky and Lederman 1997). The micropin array is also used for tactile displays. Such a device enables the provision of a teletaction and communication aid for blind individuals (Moy, Wagner, and Fearing 2000; Kawai and Tomita 2000). It has the ability to convey texture or 2D geometry (Burdea 1996).

A micropin array looks similar to an object-oriented-type force display, but it can only create the sensation of skin. The stroke distance of each pin is short, so the user cannot feel the 3D shape of a virtual object directly. The major role of a tactile display is to convey a sense of fine texture of an object's surface. Latest research on tactile displays focuses on selective stimulation of mechanoreceptors of the skin. As mentioned in Section 9.2.1, there are four types of mechanoreceptors in the skin: (1) Merkel disks, (2) Ruffini capsules, (3) Meissner corpuscles, and (4) Pacinian corpuscles. By stimulating these receptors selectively, various tactile sensations such as roughness or slip can be presented. Micro–air jets (Asamura, Yokoyama, and Shinoda 1999) and microelectrode arrays (Kajimoto et al. 1999) are used for selective stimulation. Notes 9.3–9.5 describe four types of finger/hand haptics.

9.3 TECHNOLOGIES IN FINGER/HAND HAPTICS: MANIPULATOR

9.3.1 Exoskeleton

Exoskeleton is one of the typical forms of haptic interfaces. Figure 9.6 shows the detailed view of an exoskeleton that is introduced in Section 9.2.2.1 (Iwata 1990).

A force sensation contains six-dimensional information, comprising 3D force and 3D torque. The core element of a force display is a six-DOF parallel manipulator. The typical design feature of parallel manipulators is an octahedron called "Stewart platform." In this mechanism, a top triangular platform and a base triangular platform are connected by six length-controllable cylinders. This compact hardware has the ability to carry a large payload. The structure, however, has some practical disadvantages with respect to its small working volume and its lack of backdrivability (reduction of friction) of the mechanism. In our system, three sets of parallelogram linkages (pantograph) are employed instead of linear actuators. Each pantograph is driven by two direct current (DC) motors. Each motor is powered by a pulse-width-modulation (PWM) amplifier. The top end of the pantograph is connected with a vertex of the top platform by a spherical joint. This mechanical configuration has the same advantages as an octahedron mechanism. The pantograph mechanism improves the working volume and backdrivability of the parallel manipulator. The inertia of moving parts of the manipulator is so small that compensation is not needed.

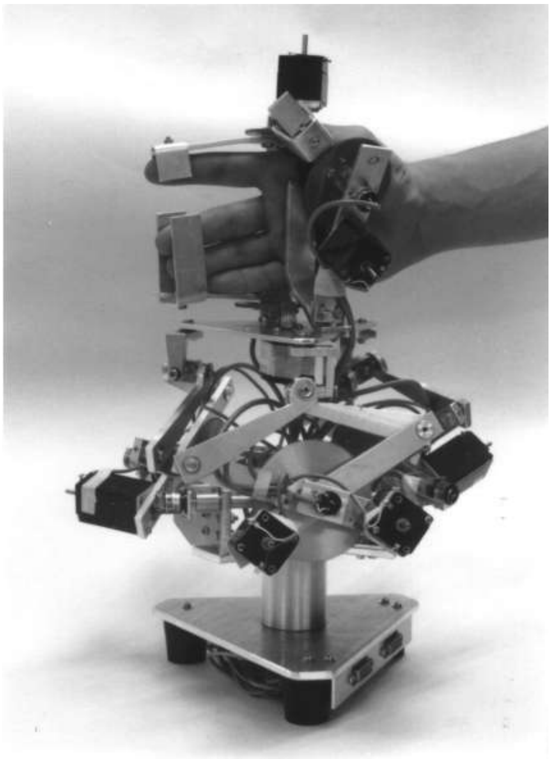


FIGURE 9.6 Mechanism of a desktop force display.

The working space of the center of the top platform is a spherical volume whose diameter is approximately 30 cm. Each joint angle of the manipulator is measured by potentiometers. Linearity of the potentiometers used is 1%. The maximum payload of the manipulator is 2.3 kg, which is more than that of a typical human hand.

The top platform of the parallel manipulator is fixed on the palm of the operator by a U-shaped attachment, which enables the operator to move his or her hand and fingers independently. Three actuators are set coaxially with the first joint of the thumb, forefinger, and middle finger of the operator. The last three fingers work together. It is noted that DC servomotors are employed for each actuator.

9.3.2 Tool-Handling-Type Haptic Interface

Users of exoskeletons feel troublesome when they put on or off these devices. This disadvantage obstructs the practical

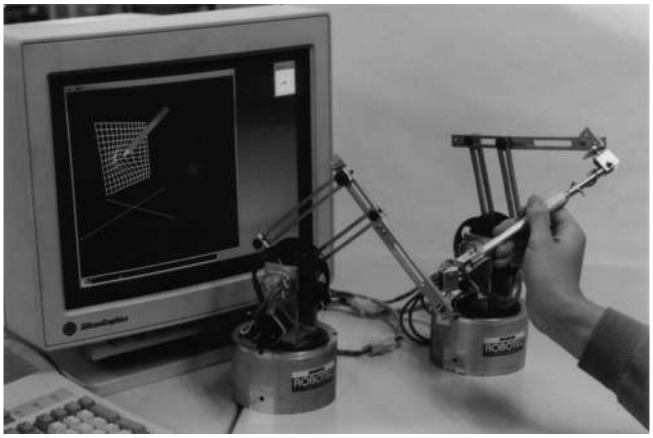


FIGURE 9.7 Pen-based force display.

use of force displays. Tool-handling-type is a method of implementation of force display without using a glovelike device. A pen-based force display is proposed as an alternative device (Iwata 1993). A six-DOF force reflective master manipulator, which has pen-shaped grip, was developed. Users are familiar with a pen in their everyday life. Most of the human intellectual work is done with a pen. People use spatulas or rakes for modeling solid objects. These devices have stick-shaped grips similar to that of a pen. In this aspect, a pen-based force display is easily applied to the design of 3D shapes.

Human hand has the ability of six-DOF motion in 3D space. In case a six-DOF master manipulator is built using serial joints, each joint must support the weight of its upper joints. These characteristics result in large hardware for the manipulator. We use parallel mechanism in order to reduce the size and weight of a manipulator. The pen-based force display employs two three-DOF manipulators. Both ends of the pen are connected to these manipulators. The force display has a total of six DOFs. A three-DOF force and a three-DOF torque are applied at the pen. An overall view of the force display is shown in Figure 9.7. Each three-DOF manipulator is composed of pantograph links. By this mechanism, the pen is kept free from the weight of the actuators.

Figure 9.8 shows the mechanical configuration of the force display. Joints MA1, MA2, MA3, MB1, MB2, and MB3 are equipped with DC motors and potentiometers. Other joints move passively. The positions of joints A and B are measured by potentiometers. A 3D force vector is applied at the joints A and B. Joint A determines position of the pen point, and joint B determines orientation of the pen. The working space of the pen point is a part of a spherical volume whose diameter is 44 cm. The rotational angle around the axis of the pen is determined by the distance between the joints A and B. A screw-motion mechanism converts rotational motion of the pen into translation along the distance between joints A and B.

Applied force and torque on the pen is generated by a combination of forces at the points A and B. In case these forces have the same direction, translational force is applied to the user's hand. If the directions of the forces are reverse to each other, a torque around the yaw axis or the pitch axis is generated. If two forces are opposite to each other, a torque around the roll axis is generated by the screw-motion mechanism.

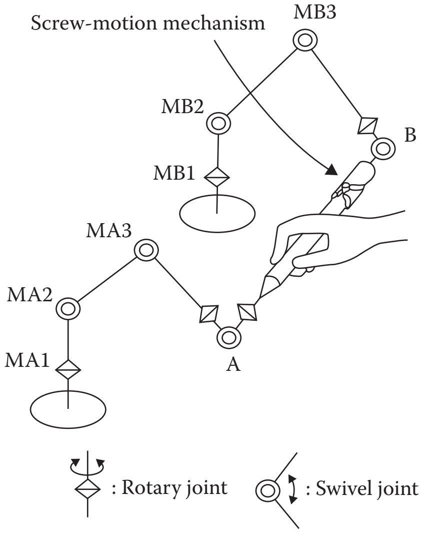


FIGURE 9.8 Mechanical configuration of a pen-based force display.

9.4 TECHNOLOGIES IN FINGER/HAND HAPTICS: OBJECT-ORIENTED-TYPE HAPTIC INTERFACE

9.4.1 Basic Idea of FEELEX

The author demonstrated haptic interfaces to a number of people and found that some of them were unable to fully experience virtual objects through the medium of synthesized haptic sensation. There seem to be two reasons for this phenomenon: First, these haptic interfaces only allow users to touch the virtual object at a single point or at a group of points. These contact points are not spatially continuous due to the hardware configuration of the haptic interfaces. The user feels a reaction force through a grip or a thimble. Exoskeletons provide more contact points, but these are achieved by using Velcro bands attached to a specific part of the user's fingers, which are not continuous. Therefore, these devices cannot recreate a natural interaction sensation when compared with manual manipulation in the real world.

The second reason why some people fail to perceive the sensation is related to a combination of visual and haptic displays. A visual image is usually combined with a haptic interface by using a conventional cathode-ray tube or projection screen. Thus, the user receives visual and haptic sensations through different displays and has to integrate the visual and haptic images in his or her brain. Some users, especially elderly people, face difficulty in this integration process.

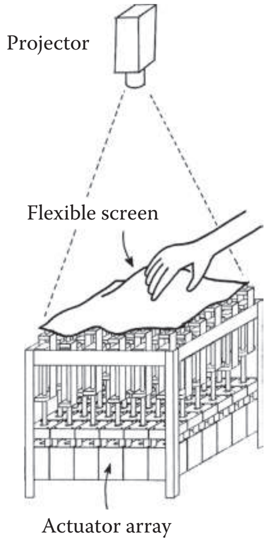


FIGURE 9.9 Basic design of FEELEX.

Considering these problems, new interface devices have been developed. The project is named FEELEX. The word FEELEX is derived from a conjunction of "feel" and "flex." The major goals of this project are as follows:

- To provide a spatially continuous surface that enables users to feel virtual objects using any part of the fingers or even the whole palm
- To provide visual and haptic sensations simultaneously using a single device that does not oblige a user to wear any extra apparatus

A new configuration of visual/haptic display was designed to achieve these goals. Figure 9.9 illustrates the basic concept of the FEELEX. The device comprises a flexible screen, an array of actuators, and a projector. The flexible screen is deformed by the actuators in order to simulate the shape of virtual objects. An image of the virtual objects is projected onto the surface of the flexible screen. Deformation of the screen converts the 2D image from the projector into a solid image. This configuration enables users to touch the image directly using any part of their hand. The actuators are equipped with force sensors to measure the force applied by a user. The hardness of a virtual object is determined by the relationship between the measured force and its position on the screen. If the virtual object is soft, a large deformation is caused by a small applied force.

9.4.2 Design Specification and Implementation of Prototypes

9.4.2.1 The FEELEX 1

The FEELEX 1, developed in 1997, was designed to enable double-handed interaction using the whole of the palm. Therefore, the optimum size of the screen was determined to be 24 cm × 24 cm. The screen is connected to a linear actuator array that deforms its shape. Each linear actuator comprises a screw mechanism driven by a DC motor. The screw mechanism converts the rotation of an axis of the motor to the linear motion of a rod. The motor must generate both motion and a reaction force on the screen. The diameter of the smallest motor that can drive the screen is 4 cm. Therefore, a 6 × 6 linear actuator array can be set under the screen. The deformable screen is made of a rubber plate and a white nylon cloth. The thickness of the rubber plate is 3 mm. Figure 9.10 shows an overall view of the device.

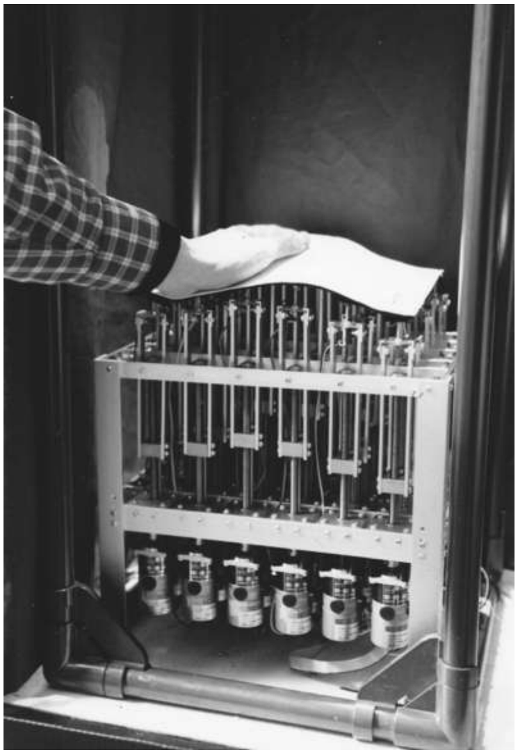


FIGURE 9.10 Overall view of FEELEX 1.

The screw mechanism of the linear actuator has a self-lock function that maintains its position when the motor power is off. A hard virtual wall is difficult to simulate using tool-handling-type force displays. Considerable motor power is required to generate the reaction force from the virtual wall, which often leads to uncomfortable vibrations. The screw mechanism is free from this problem. A soft wall can be represented by the computer-controlled motion of linear actuators based on the data from the force sensors. A force sensor is set at the top of each linear actuator. Two strain gauges are used as a force sensor. The strain gauge detects small displacements of the top end of the linear actuator caused by the force applied by the user. The position of the top end of the linear actuator is measured by an optical encoder connected to the axis of the DC motor. The maximum stroke of the linear actuator is 80 mm, and the maximum speed is 100 mm/s.

The system is controlled via a personal computer. The DC motors are interfaced by a parallel input/output unit, and the force sensors are interfaced by an alternating current (AC) to DC (A/D) converter unit. The force sensors provide interaction with the graphics. Position and strength of the force applied by the user are detected by a 6 × 6 sensor array. The graphics projected onto the flexible screen are changed according to the measured force.

9.4.2.2 The FEELEX 2

The FEELEX 2 is designed to improve the resolution of the haptic surface. In order to determine the resolution of the linear actuators, we considered the situation in which a medical doctor palpates a patient. The results of interviewing several medical doctors proved that they usually recognized a tumor using their index finger, middle finger, and third finger. The size of a tumor is perceived by comparing it to the width of their fingers, that is, two-finger large or three-finger large tumors. Thus, the distance between the axis of the linear actuators should be smaller than the width of a finger. Considering the aforementioned condition, the distance is set to 8 mm. This 8-mm resolution enables the user to hit at least one actuator when he or she touches any arbitrary position on the screen. The size of the screen is 50 mm × 50 mm, which allows the user to touch the surface using three fingers.

In order to realize 8-mm resolution, a piston–crank mechanism is employed for the linear actuator. The size of the motor is much larger than 8 mm, so the motor should be placed at a position offset from the rod. The piston–crank mechanism can easily achieve this offset position. Figure 9.11 illustrates the mechanical configuration of a linear actuator. A servomotor from a radio-controlled car is selected as the actuator. The rotation of the axis of the servomotor is converted to the linear motion of the rod by a crankshaft and a linkage. The stroke of the rod is 18 mm, and the maximum speed is 250 mm/s. The maximum torque of the servomotor is 3.2 kg · cm, which applies a 1.1 kg force at the top of each rod. This force is sufficient for palpation using fingers.

The flexible screen is supported by 23 rods, and the servomotors are set remotely from the rods. Figure 9.12 shows an overall view of FEELEX 2. The 23 separate sets of piston–crank mechanisms can be seen in the figure.

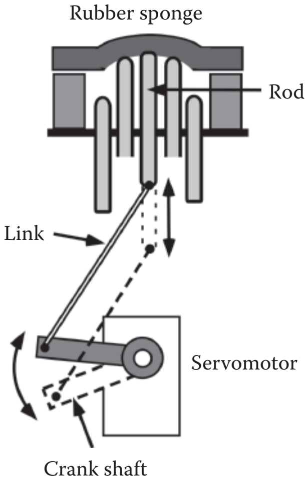


FIGURE 9.11 The piston–crank mechanism.

Figure 9.13 shows the top end of the rods. The photograph was taken when the flexible screen was off. The diameter of each rod is 6 mm. A strain gauge cannot be put on top of the rod because of its small size. Thus, the electric current going to each servomotor is measured to sense the force. The servomotor generates a force to maintain the position of the crankshaft. When a user applies a force to the rod, the electric current on the motor increases to balance the force applied. The relationship between the applied force and the electric current is measured. The force applied at the top of the rods is calculated using data from the electric current sensor. The resolution of the force-sensing capability is 40 gf.

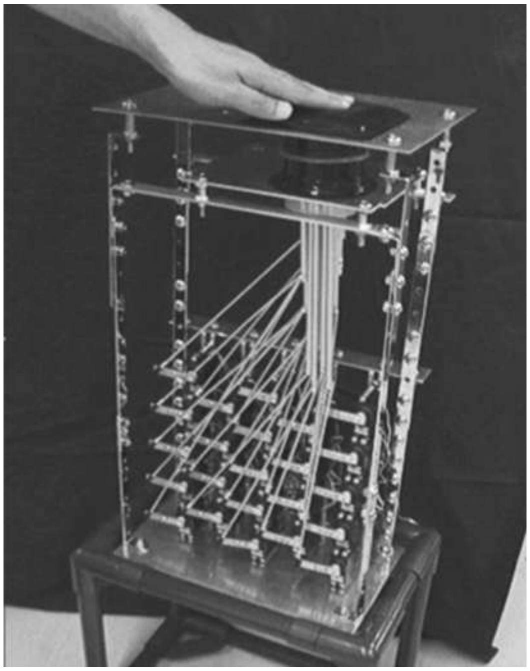


FIGURE 9.12 Overall view of FEELEX 2.

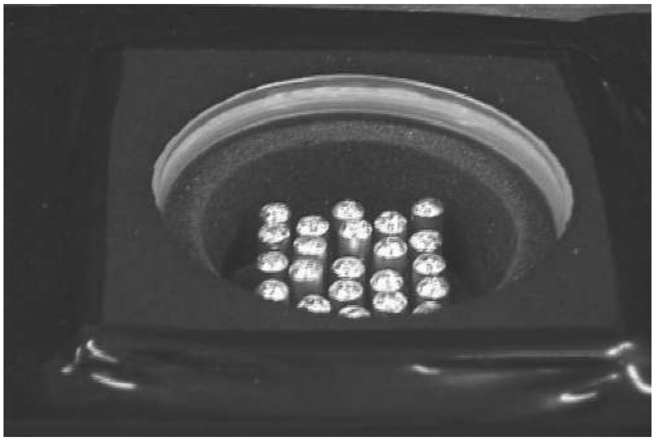


FIGURE 9.13 Top end of the rods conforming haptic surface.

9.4.3 Characteristics of FEELEX

The performance of existing haptic interfaces is usually represented by the dynamic range of force, impedance, inertia, friction, and so on. However, these parameters are crucial only while the device is attached to a finger or the hand. In the case of a tool-handling-type haptic interface or the exoskeleton, the devices move with the hand even though the user does not touch the virtual objects. Therefore, inertia or friction degrades the usability and dynamic range of force determines the quality of the virtual surface. On the other hand, FEELEX is entirely separate from the user's hand; so its performance is determined by the resolution and speed of the actuators. The resolution of the actuator corresponds to the smoothness of the surface, and its speed determines the motion of the virtual object. Compared with FEELEX 1, FEELEX 2 has improved resolution and motion speed. Each actuator of FEELEX 2 has a stroke rate of up to 7 Hz, which can simulate the motion of a very fast virtual object. The rod pushes the rubber sponge so that the user feels as if the object is pulsating. It is of interest that 7 Hz is much faster than the human pulse rate.

The major advantage of FEELEX is that it allows natural interaction using only the bare hand. In SIGGRAPH '98, 1992 subjects spontaneously enjoyed the haptic experience. One of the subject contents of the FEELEX 1 system, known as Anomalocaris, was selected as a long-term exhibition at the Ars Electronica Center (Linz, Austria). The exhibition has been popular among visitors, especially children.

Another advantage of FEELEX is safety. The user of FEELEX does not wear any special equipment while the interaction is taking place. The exoskeleton and tool-handling-type force displays have control problems in their contact surface for the virtual objects. Vibration or unwanted forces can be generated back to the user, which is sometimes dangerous. The contact surface of the FEELEX system is physically generated, so it is free from such control problems.

The major disadvantage of FEELEX is the degree of difficulty present in its implementation. It requires a large number of actuators that must be controlled simultaneously. The drive mechanism of the actuator must be robust enough for rough manipulation. Since FEELEX provides a feeling of natural interaction, some of the users apply large forces. Our exhibit at the Ars Electronica Center suffered from an overload of actuators.

Another disadvantage of FEELEX is its limitation in presenting the shape of objects that can be displayed. Current prototypes cannot present a sharp edge on a virtual object. Furthermore, the linear actuator array can simulate only the front face of objects. Some of the participants of the Anomalocaris demonstration wanted to touch the rear of the creature, but an entirely new mechanism would be required in order to also simulate the reverse side of an object.

9.4.4 Volumetric Object-Oriented-Type Haptic Interface

9.4.4.1 Basic Design of Volflex

In order to present the side or backside of a virtual object, we designed a volumetric object-oriented haptic interface. Volflex is a new haptic interface that provides the user a physical 3D surface for interaction. The device comprises a group of air balloons. The balloons fill the interaction surface (Figure 9.14). They are arranged in a body-centered cubic lattice. A tube is connected to each balloon. Volume of each balloon is controlled by an air cylinder. The tubes are connected to each other by springs. This mechanical flexibility enables arbitrary shape of the interaction surface. Each air cylinder is equipped with a pressure sensor that detects the force applied by a user. According to the pressure data, the device is programmed to perform like clay (Figure 9.15). Unlike real clay, Volflex allows a user to "undo" an operation.

A projector is set above the balloons. The image is projected on the surface of the device, not on the user's hand. We developed a mechanical rotary shutter that separates the projector and the camera. The camera captures the user's hand, which is eliminated from the projected image.

9.4.4.2 Virtual Clay Volflex

Virtual clay is one of the ultimate goals of the interactive technique of 3D graphics. Digital tools for 2D paint comprise mature technology. On the other hand, tools for 3D shape manipulation are currently in a preliminary stage of development. Shape design of 3D objects is one of the major application areas of haptic interfaces. Shape design of 3D objects requires good sensation of haptics.

Volflex provides an effective interface device for the manipulation of virtual clay by using a lattice of air balloons.

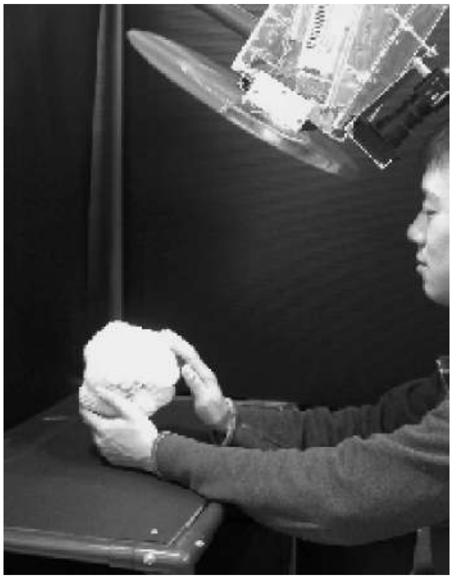


FIGURE 9.14 Overall view of Volflex.

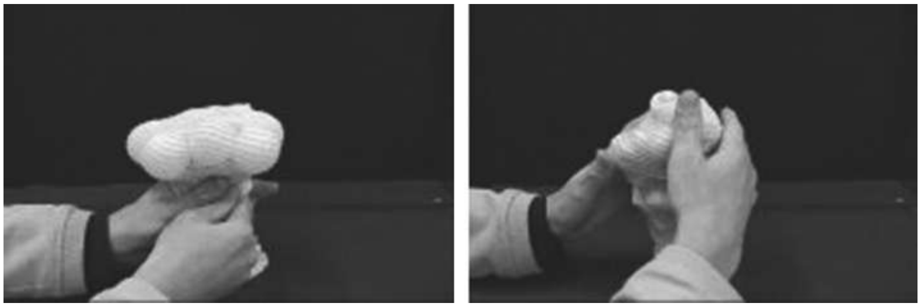


FIGURE 9.15 Examples of deformation.

Two-dimensional paint tools are popular and digital pictures are easy to draw. Volflex is a new digital tool for making 3D shapes. It has the potential to bring about a revolution in methods of industrial design. Designers use their palms or the joints of their fingers to deform a clay model when carrying out rough design tasks. Volflex has the ability to support such natural manipulations.

Volflex is not only a tool for 3D shape designing but also an interactive artwork in itself. A physical property of a virtual object can be designed by programming the controllers of the balloons. It is also possible to design a projected image. The combination of haptic and visual displays provides a new platform for interactive sculpture.

9.5 TECHNOLOGIES IN FINGER/HAND HAPTICS: REMOTE HAPTICS

In the real world, we usually touch real objects that are placed on a reachable area and we can notice their properties by touching them. However, we cannot touch real objects that are placed on an untouchable area. In daily life, there are many untouchable objects such as objects in the showcase of a museum. However, if we can touch them we can learn many things from such valuable objects. Therefore, the objective of this study is to realize a feeling of touching untouchable objects by using a haptic interface that has no need to premeasure their shapes, although we do not touch them directly.

Figure 9.16 illustrates the basic method of remote haptics. As a real-time shape measurement device, a laser range finder (LRF) is used. It can measure the distance to a front object in less than 1 millisecond. Since laser light can pass through glass, the LRF can measure the distance to an object even when the object is placed beyond a glass. Therefore, we can measure the distance to an untouchable object without there being any prerecorded data.

For using a haptic interface at a museum, the interface should be a mobile one. Two types of haptic interfaces can be considered for this purpose: (1) One is a wearable type of haptic interface and (2) the other is a handheld type. It is difficult to attach and remove a wearable type of haptic interface, and if it is attached for a long time the user will get fatigued. On the other hand, handheld types of haptic interfaces are easier to use, since they are easy to attach and remove when the user gets fatigued. Hence, handheld-type haptic interfaces can be applied for this purpose. In addition, force feedback for the whole hand or multifinger is not practical since the apparatus is complex and heavy. As a first step, a one-DOF haptic interface was used in our study. For a haptic rendering, the position of a virtual object is determined based on the distance data from an LRF. When a user's finger penetrates the virtual object, reaction force, which is proportional to penetration depth from the surface of the virtual object, is applied to the user's finger. In this case, when the user presses the interface to a glass case and moves it freely on the surface of the glass, the user can feel the reaction force, which is proportional to the "depth information" from the LRF. Then the user can reconstruct the shape information of a given object by integrating the depth information from the interface and the "position sense of user's hand." Figure 9.17 shows the overall view of the system. This system can be applied for educational purposes in understanding many characteristics of valuable exhibits in museums and for quality inspection of engineering products.

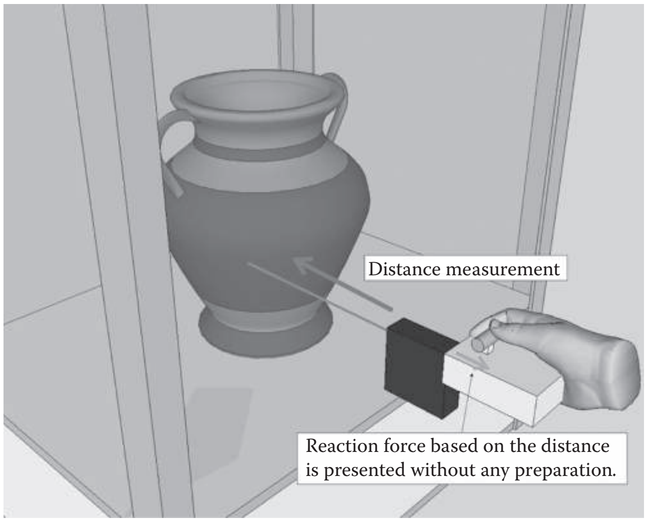

Distance measurement
Reaction force based on the distance is presented without any preparation.

FIGURE 9.16 Basic method of remote haptics.

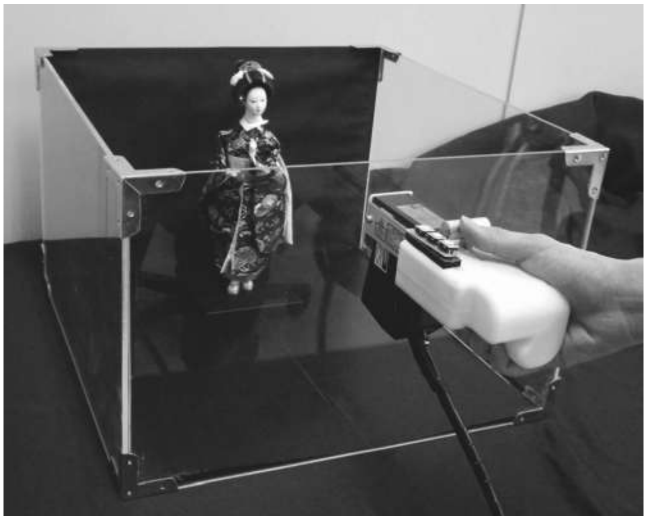


FIGURE 9.17 Overall view of the remote haptics system.

9.6 TECHNOLOGIES IN FULL-BODY HAPTICS

9.6.1 Treadmill-Based Locomotion Interface

9.6.1.1 Basic Design of the Torus Treadmill

A key principle of treadmill-based locomotion interfaces is to make the floor move in a direction opposite to that of the walker. The motion of the floor cancels the displacement of a walker in the real world. The major challenge of a treadmill-based locomotion interface is to allow the walker to change his or her direction. An omnidirectional active floor enables a virtually infinite area. In order to realize an infinite walking area, geometric configuration of an active floor must be chosen. A closed surface driven by actuators has the ability to create an unlimited floor. The following requirements for implementing a closed surface must be considered:

- The walker and the actuators must be placed outside the surface.
- The walking area must be a plane surface.
- The surface must be made of a material that stretches very little.

A closed surface in general is a surface with holes. If the number of holes is zero, the surface is a sphere. The sphere is the simplest infinite surface. However, the walking area of a sphere is not a plane surface. A very large diameter is required to make the surface plane on a sphere, which restricts the implementation of a locomotion interface.

A closed surface with one hole like a doughnut is called a "torus." A torus can be implemented by a group of belts. These belts make a plane surface for the user to walk on. A closed surface with more than one hole cannot make a plane walking surface. Thus, torus is the only form suitable for a locomotion interface.

9.6.1.2 Mechanism and Performance

The Torus Treadmill is implemented by a group of belts connected to each other. The Torus Treadmill is realized by these belts (Iwata 1999). Figures 9.18 and 9.19 illustrate the basic structure of a Torus Treadmill. The Torus Treadmill employs 12 treadmills. These treadmills move a walker along an *x* direction. The 12 treadmills are connected side by side and driven in a perpendicular direction. This motion moves the walker along a *y* direction.

Figure 9.20 shows the overall view of the apparatus. A total of 12 treadmills are connected to four chains and mounted on four rails. The chain drives the walker along

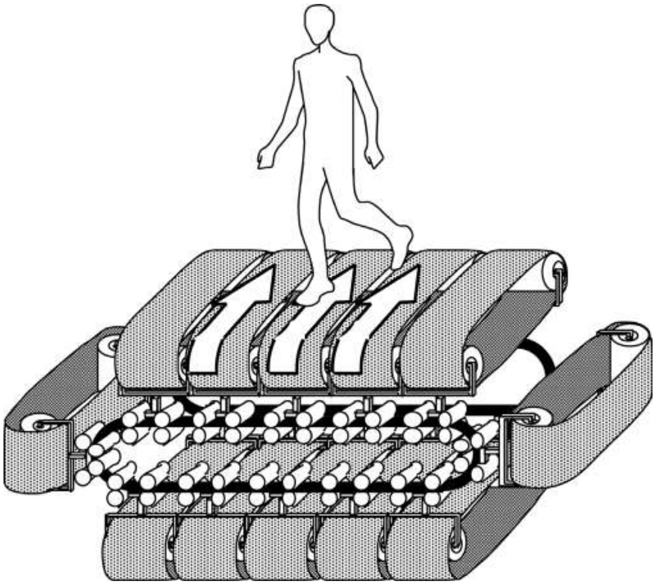


FIGURE 9.18 Structure of a Torus Treadmill (*x* motion).

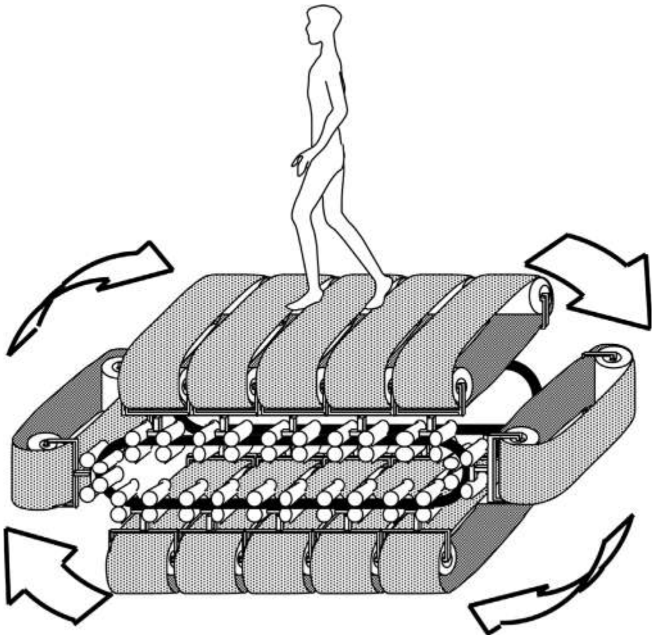


FIGURE 9.19 Structure of a Torus Treadmill (*y* motion).

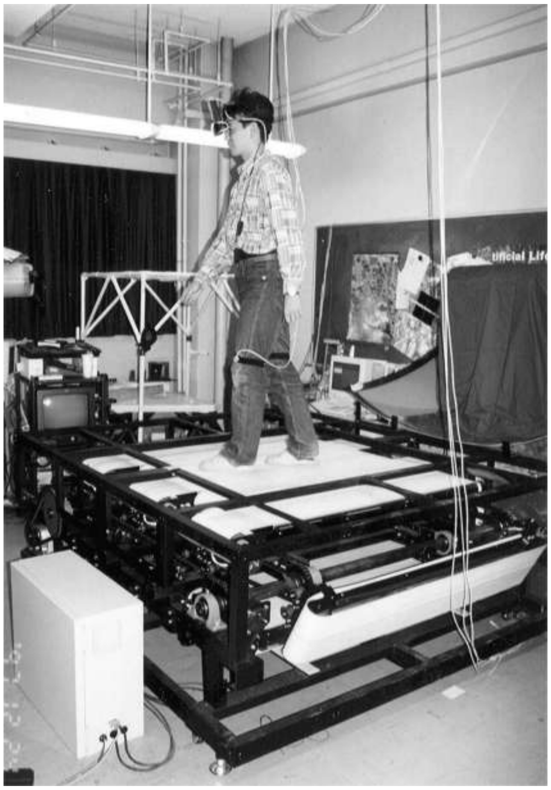


FIGURE 9.20 Overall view of the Torus Treadmill.

the *y* direction. The rail supports the weight of the treadmills and the walker. An AC motor is employed to drive the chains. The power of the motor is 200 W and it is controlled by an inverter. The maximum speed of rotation is 1.2 m/s. The maximum acceleration is 1.0 m/s^2. The deceleration caused by friction is 1.5 m/s^2. Frequency characteristics are limited by a circuit protector of the motor driver. The maximum switching frequency is 0.8 Hz.

Each treadmill is equipped with an AC motor. In order to shorten the length of the treadmill, the motor is put underneath the belt. The power of each motor is 80 W and it is controlled by an inverter. The maximum speed of each treadmill is 1.2 m/s. The maximum acceleration is 0.8 m/s^2. The deceleration caused by friction is 1.0 m/s^2. The width of each belt is 250 mm and the overall walkable area is 1 m × 1 m.

A problem with this mechanical configuration is the gap between the belts in the walking area. In order to minimize the gap, we put a driver unit of each treadmill alternatively. The gap is only 2-mm wide in this design.

9.6.1.3 Control Algorithm of the Torus Treadmill

A scene of the virtual space is generated corresponding to the results of motion tracking of the feet and head of the walker. The motion of the feet and head is measured by a Polhemus FASTRACK. The device measures six-DOF motion. The sampling rate of each point is 20 Hz. A receiver is attached to each knee. We cannot put sensors near the motion floor because a steel frame distorts magnetic field. The length and direction of a step is calculated by the data from those sensors. The user's viewpoint in virtual space moves in accordance with the length and direction of his or her steps.

To keep the walker in the center of the walking area, the Torus Treadmill must be driven in correspondence with the walker. A control algorithm is required to achieve safe and natural walking. From our experience from the Virtual Perambulator project, the walker should not be connected to a harness or mechanical linkages since such devices restrict the motion and inhibit natural walking. The control algorithm of the Torus Treadmill must be safe enough to allow removal of the harness from the walker. At the final stage of the Virtual Perambulator project, we succeeded in removing the harness using a hoop frame. The walker can freely walk and turn around in the hoop, which supports the walker's body while he or she slides the feet. We simulated the function of the hoop in the control algorithm of the Torus Treadmill by putting a circular deadzone in the center of the walking area. If the walker steps out of this area, the floor moves in the opposite direction so that the walker is carried back into the deadzone.

9.6.2 Footpad-Based Locomotion Interface

9.6.2.1 Methods of Presentation of Uneven Surfaces

One of the major research issues in the field of locomotion interfaces is the presentation of uneven surfaces. Locomotion interfaces are often applied for the simulation of buildings or urban spaces. Those spaces usually include stairs. A walker should be provided a sense of climbing up or going down these stairs. Some applications of locomotion interfaces, such as training simulators or entertainment facility, rough terrain should be presented.

The presentation of a virtual staircase was tested in the early stage of the Virtual Perambulator project (Iwata and Fujii 1996). A staircase is a typical example of a rough terrain. A string is connected to the roller skate of each foot. The string is pulled by a motor. When the walker climbs up a stair, the forward foot is pulled up. When the walker goes down a stair, the backward foot is pulled up. However, this method was not successful because of instability.

Later, a six-DOF motion platform was applied to the final version of the Virtual Perambulator, where a user walked in a hoop frame. The walker stood on the top plate of the motion platform. Pitch and heave motions of the platform were used. When the walker stepped forward to climb up a stair, the pitch angle and vertical position of the floor increased. After finishing the climbing motion, the floor went back to the neutral position. When the walker stepped forward to go down a stair, the pitch angle and vertical position of the floor decreased. This inclination of the floor was intended to present height difference between the feet. The heave motion was intended to simulate vertical acceleration. However, this method failed in simulating the stairs. The major reason was that the floor was flat.

A possible method for creating height difference between the feet is application of two large manipulators. The BiPort is a typical example. A four-DOF manipulator driven by hydraulic actuators is connected to each foot. A major problem of this method involves how the manipulators trace the turning motion of a walker. When the walker turns around, two manipulators interfere with each other.

The Torus Treadmill provides natural turning motion. The walker on the Torus Treadmill can physically turn about on the active floor. Turning motion using the feet makes a major contribution to human spatial recognition performance. Vestibular and proprioceptive feedback is essential to the sense of orientation (Iwata and Yoshida 1999). The Torus Treadmill can be modified for the simulation of uneven surfaces. If we install an array of linear actuators on each treadmill, an uneven floor can be realized by controlling the length of each linear actuator. However, this method is almost impossible to implement, because a very large number of linear actuators are required to cover the surface of the torus-shaped treadmills and the control signal for each actuator must be transmitted wirelessly.

9.6.2.2 Basic Design of the GaitMaster

A new locomotion interface that simulates omnidirectional uneven surfaces has been designed. The device is named GaitMaster. The core elements of the device are two six-DOF motion bases mounted on a turntable. Figure 9.21 illustrates the basic configuration of the GaitMaster.

A walker stands on the top plate of a motion base. Each motion base is controlled to trace the position of the foot. The turntable is controlled to trace the orientation of the walker. The motion of the turntable removes interference between the two motion bases.

The *x* and *y* motions of the motion base traces the horizontal position of the feet and cancel its motion by moving in the opposite direction of the feet. The rotation around the yaw axis traces the horizontal orientation of the feet. The *z* motion traces vertical position of the feet and cancels their motion. The rotation around the roll and pitch axes simulates the inclination of a virtual surface.

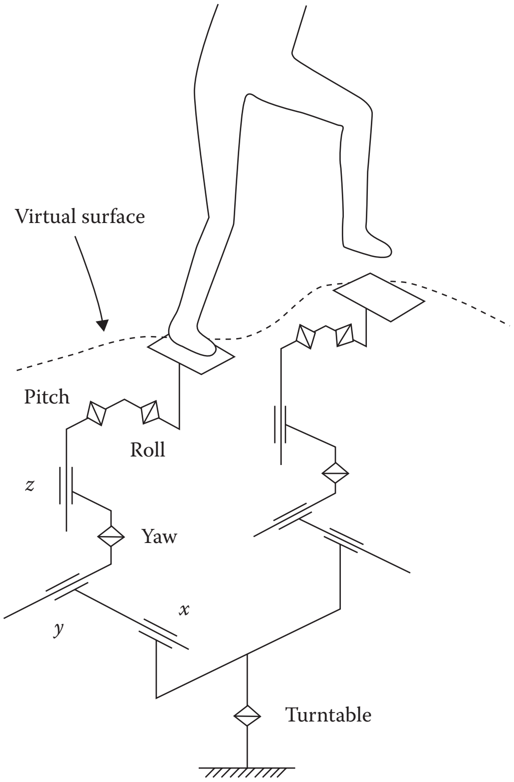


FIGURE 9.21 Basic design of the GaitMaster.

9.6.2.3 Control Algorithm of the GaitMaster

The control algorithm must keep the position of the walker at the neutral position of the GaitMaster. In order to maintain the position, the motion of motion platforms must cancel the motion of the feet. The principle of cancellation is explained in the following four steps:

1. Suppose the right foot of the walker is at the forward position and left foot is at the backward position while walking.
2. When the walker puts his or her left foot forward, the weight of the walker falls on the right foot.
3. The motion platform of the right foot goes backward in accordance with the displacement of the left foot so that the central position of the walker is maintained.
4. The motion platform of the left foot follows the position of the left foot. When the walker finishes stepping forward, the motion platform supports the left foot.

If the walker climbs up or goes down a flight of stairs, a similar procedure can be applied. The vertical motion of the feet is canceled using the same aforementioned principle. The vertical displacement of the forward foot is canceled in accordance with the motion of the backward foot, so that the central position of the walker is maintained at the neutral height. Figure 9.22 illustrates the method of canceling the climbing-up motion.

The turntable rotates so that the two motion platforms can trace the rotational motion of the walker. If the walker changes the direction of walking, the turntable rotates to trace the orientation of the walker. The turntable orientation

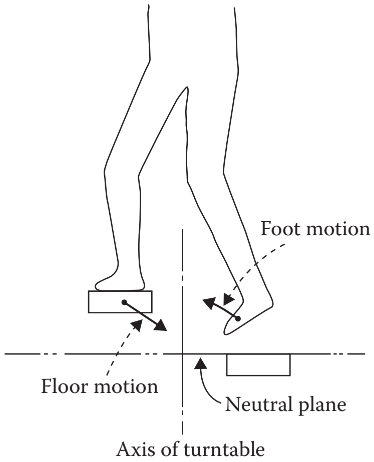


FIGURE 9.22 Method of canceling the climbing-up motion of a walker.

is determined according to the direction of the feet. The turntable rotates so that its orientation is at the middle of the feet. The walker can physically turn around on the GaitMaster using this control algorithm of the turntable.

9.6.2.4 Prototype GaitMaster

Figure 9.23 shows the overall view of the prototype GaitMaster. In order to simplify the mechanism of the motion platform, the surface of the virtual space was defined as sets of plane surfaces. Most of the buildings or urban spaces can be simulated without inclination of the floor. Thus, we can neglect the roll and pitch axes of motion platforms. Each platform of the prototype GaitMaster comprises three linear actuators atop of which a yaw joint is mounted. We disassembled a six-DOF Stewart platform and made two *xyz* stages. Three linear guides are applied to support the orientation of the top plate of the motion platform. The payload of each motion platform is approximately 150 kg. A rotational joint around the yaw axis is mounted on each motion platform. The joint is equipped with a spring that moves the feet to the neutral direction.

A turntable is developed using a large direct drive (DD) motor. The maximum angular velocity is 500°/s. A three-DOF goniometer is connected to each foot. The goniometer measures back-and-forth and up-and-down motion as well as the yaw angle. The control algorithm mentioned in Section 9.6.2.3 was implemented and it succeeded in the presentation of virtual staircases.

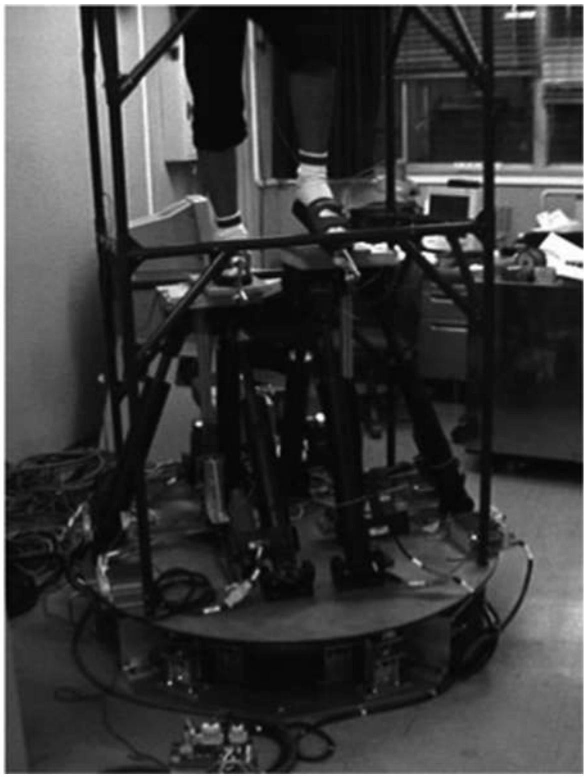


FIGURE 9.23 Overall view of the prototype GaitMaster.

9.6.3 Robot-Tile-Based Locomotion Interface

9.6.3.1 CirculaFloor Project

Locomotion interfaces often require bulky hardware, since they have to carry a user's whole body. Also, the hardware is not easy to reconfigure to improve its performance or to add new functions. Considering these issues, the goals of the CirculaFloor project are as follows:

To develop a compact hardware for the creation of an infinite surface for walking: The major disadvantage of existing locomotion interfaces is the difficulty in installation. We need to solve this problem for a demonstration at SIGGRAPH.

To develop scalable hardware architecture for future improvement of the system: Another disadvantage of existing locomotion interfaces is the difficulty faced when improving the system. We have to design a new hardware architecture that allows us to easily upgrade the actuation mechanism or to add new mechanisms for the creation of uneven surfaces.

In order to achieve these goals, we designed a new configuration for a locomotion interface by using a set of omnidirectional movable tiles. Each tile is equipped with a holonomic mechanism that achieves omnidirectional motion. An infinite surface is simulated by the circulation of movable tiles. The motion of the feet is measured by position sensors. The tile moves opposite to the measured direction of the walker so that the motion of the step is canceled. The position of the walker is fixed in the real world by this computer-controlled motion of the tiles. The circulation of the tiles has the ability to cancel the displacement of the walker in an arbitrary direction. Thus, the walker can freely change his or her direction while walking. Figure 9.24 shows an overall view of CirculaFloor.

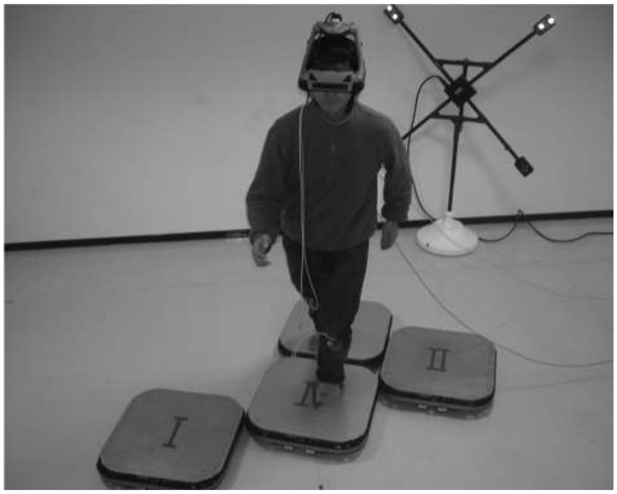


FIGURE 9.24 Overall view of the CirculaFloor project.

The CirculaFloor is a new method that takes advantage of both the treadmill and the footpad. It creates an infinite omnidirectional surface by using a set of movable tiles. The combination of tiles provides a sufficient area for walking and, thus, precision tracing of foot position is not required. It has the potential to create an uneven surface by mounting an up-and-down mechanism on each tile.

9.6.3.2 Method of Creating an Infinite Surface

The current method of circulating movable tiles is designed to satisfy the following conditions:

- Two of the movable tiles are used for pulling back the user to the center of the deadzone.
- The rest of the movable tiles are used to create a new front surface.
- These tiles are moved the shortest distances to the next destination, while they avoid colliding with other tiles.
- The control program allocates all destinations to the tiles, when the tiles reach their destinations.
- The tiles do not rotate corresponding to the walking direction to simplify the algorithm.

According to the aforementioned conditions, the circulation method is varied corresponding to the walking direction. Three modes, "alternating circulation," "unidirectional circulation," and "crossed circulation," are designed corresponding to the direction (Figure 9.25); representative motions of each mode are illustrated in Figures 9.26 through 9.28:

Alternating circulation (Figure 9.26): This mode is adopted for the directions between ±15° and ±75°–105°. The tiles used for creating a new front surface (white-colored tiles in Figure 9.26) move around to the front of the tiles for alternatively pulling back (in Figure 9.24, gray-colored tiles) from the left (path 1)/right (path 2) sides.

Unidirectional circulation (Figure 9.27): This mode is adopted for the directions between ±15°–30° and ±60°–75°. The tiles used for creating a new front surface move around to the right/left front of the tiles for pulling back with a unidirectional circulation.

Crossed circulation (Figure 9.28): This mode is adopted for directions of ±30°–60°. The tiles used for creating a new front surface move around to the left/right front (path 1) or the left/right sides (path 2) of the tiles for pulling back.

When a user of the CirculaFloor switches his or her walking direction, the control program calculates the nearest phase of each tile by using a template-matching technique corresponding to the new direction. Then the tiles take the shortest way to their destinations.

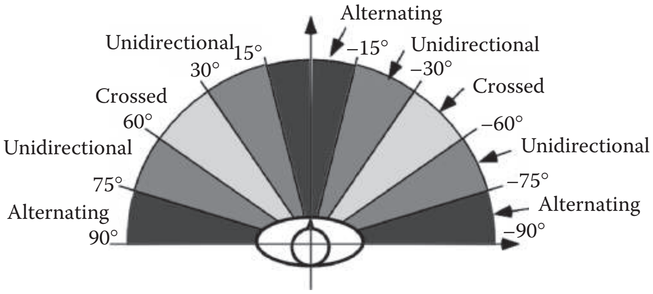


FIGURE 9.25 Pulling-back modes corresponding to walking directions.

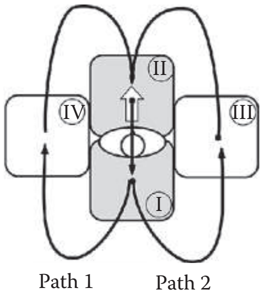


FIGURE 9.26 Circulation of movable tiles in alternating mode.

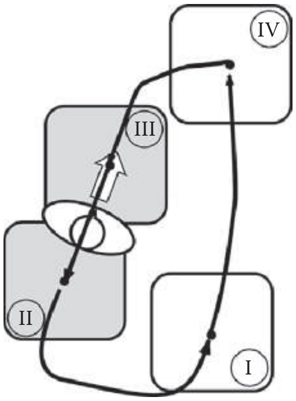


FIGURE 9.27 Circulation of movable tiles in unidirectional mode.

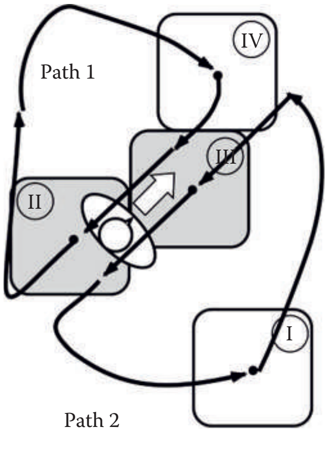


FIGURE 9.28 Circulation of movable tiles in crossed circulation mode.

9.7 APPLICATION AREAS FOR HAPTIC INTERFACE

9.7.1 Application Areas for Finger/Hand Haptics

9.7.1.1 Medicine

Medical applications for haptic interfaces are currently growing rapidly. Various surgical simulators have been developed using tool-handling-type force displays. We developed a simulator for laparoscopic surgery using the HapticMaster. Simulator software using PHANToM are commercially available.

Palpation is typically used in medical examinations. The FEELEX 2 is designed to be used as a palpation simulator. If we display a virtual tumor based on a computed tomography or magnetic resonance imaging image, a medical doctor can palpate the internal organs before surgery; this technique can be also applied to telemedicine. Connecting two FEELEXs together via a communication line would allow a doctor to palpate a patient remotely.

9.7.1.2 Three-Dimensional Shape Modeling

The design of 3D shapes definitely requires haptic feedback. A typical application of the tool-handling-type force display is in 3D-shape modeling. One of the most popular applications of the PHANToM system is as a modeling tool. Such a tool-handling-type force display allows a user to contact at a point, and point contact manipulation is most suited for precision modeling tasks. However, it is not effective when the modeling task requires access to the whole shape. Designers use their palm or the joints of their fingers to deform a clay model when carrying out rough design tasks. The FEELEX has the ability to support such natural manipulation.

9.7.1.3 Haptic User Interface

Today, touch screens are widely used in automatic teller machines, ticketing machines, information kiosks, and so on. A touch screen enables an intuitive user interface, although it lacks haptic feedback. Users can see virtual buttons, although they cannot feel them. This is a serious problem for a blind person. The FEELEX provides a barrier-free solution to the touch-screen-based user interface. Figure 9.13 shows an example of a haptic touch screen using FEELEX 1.

9.7.1.4 Art

Interactive art may be one of the best applications of the FEELEX system. As we discussed in Section 9.4.5, the Anomalocaris has been exhibited at a museum in Austria. It succeeded in evoking haptic interaction with many visitors. The FEELEX can be used for interactive sculptures. Although visitors are usually prohibited from touching physical sculptures, they cannot only touch sculptures based around FEELEX but also deform them.

9.7.2 Application Areas for Locomotion Interfaces

As a serious application of our locomotion interface, we are currently working with the Ship Research Laboratory to develop an "evacuation simulator" (Yamao et al. 1996). The Ship Research Laboratory is a national research institute that belongs to the ministry of transportation of Japan. Analysis of evacuation of passengers during maritime accidents is very important for ship safety. However, it is impossible to carry out experiments with human subjects during an actual disaster. Therefore, researchers introduced virtual reality tools for simulating a disaster in order to analyze the evacuation of passengers. They built a virtual ship that models the generation of smoke and the inclination of the vessel. Experiments of evacuation are carried out for constructing a mathematical model of passengers' behavior during disaster. The Torus Treadmill will be effective in such experiments.

Locomotion by walking motion is intuitive and is inevitable in studies on human behavior in virtual environments. We are applying the system to conduct research on human model of evacuation in maritime accidents. The GaitMaster can be applied to areas other than the ones for Torus Treadmill. Its application may include rehabilitation of walking or simulators for mountain climbing.

9.8 CONCLUSION AND FUTURE PROSPECTS

This chapter describes major topics under the field of haptic interfaces. A number of methods have been proposed to implement haptic interfaces. Future work in this research field will include the two issues discussed in Sections 9.8.1 and 9.8.2.

9.8.1 Safety Issues

Safety of users is an important consideration in haptic interface development. Inadequate control of actuators may injure the user. The exoskeleton and tool-handling-type force displays have control problems in their contact surface for virtual objects. Vibrations or unwanted forces can be generated and transmitted back to the user, which are sometimes dangerous. One of the major advantages of FEELEX is its safety. The user of FEELEX does not wear any special equipment while the interaction is taking place. The contact surface of FEELEX is physically generated, so it is free from control problems.

Locomotion interfaces pose much more important safety issues. The system supports the whole body of the user so that inadequate control causes major damage to the user. Specialized hardware for maintaining the safety of the walker must be developed.

9.8.2 Psychology in Haptics

There have been many findings regarding haptic sensation. Most of these are related to skin sensation, and research activities on muscle sensation are very few in number. Among these, Lederman and Klatzky's work (1987) is closely related to the design of force displays. Their latest work involves spatially distributed forces (Lederman and Klatzky 1999). They also performed an experiment involving palpation. The subjects were asked to find a steel ball placed underneath a foam rubber cover. The results showed that steel balls smaller than 8 mm in diameter decreased the score. This finding supports our specification for FEELEX 2 in which distance between the rods is 8 mm. This kind of psychological studies will support future development of haptic interfaces.

Haptics is indispensable for human interaction in the real world. However, haptics is not commonly used in the field of HCI. Although there are several commercially available haptic interfaces, they are expensive and limited in their function. Image display has a history spanning over 100 years. Today, image displays, such as television displays or movies, are used in everyday life. On the other hand, haptic interfaces have only a 10-year-old history. There are hazards to overcome before the popular use of haptic interfaces can become a reality. However, haptic interface forms a new frontier of media technology and it will definitly contribute to the betterment of human life.

REFERENCES

Asamura, N., N. Yokoyama, and H. Shinoda. 1999. A method of selective stimulation to epidermal skin receptors for realistic touch feedback. In *Proc IEEE Virtual Reality '99*, 274–81. Washington, DC: IEEE Computer Society.

Brooks Jr., F. P. 1986. A dynamic graphics system for simulating virtual buildings. In *Proceedings of the 1986 Workshop on Interactive 3D Graphics (Chapel Hill, NC, October 1986)*, 9–21. New York: ACM.

Brooks, F. P., M. Ouh-Young, J. J. Batter, and P. J. Kilpatrick. 1990. Project GROPE—haptic displays for scientific visualization. *Comput Graphics* 24(4):177–85.

Burdea, G. C. 1996. *Force and Touch Feedback for Virtual Reality*. New York: A Wiley-Interscience Publication.

Burdea, G., J. Zhuang, E. Roskos, D. Silver, and L. Langlana. 1992. A portable dextrous master with force feedback. *Presence* 1(1).

Christensen, R., J. M. Hollerbach, Y. Xu, and S. Meek. 1998. Inertial force feedback for a locomotion interface. In *Proc. ASME Dynamic Systems and Control Division*, DSC-Vol. 64, 119–26. Cambridge, MA: MIT Press.

Darken, R., W. Cockayne, and D. Carmein. 1997. The omni-directional treadmill: A locomotion device for virtual worlds. In *Proceedings of UIST '97*. New York: ACM.

Hirota, K., and M. Hirose. 1996. Simulation and presentation of curved surface in virtual reality environment through surface display. In *Proc. of IEEE VRAIS '96*. Washington, DC: IEEE Computer Society.

Iwata, H. 1990a. Artificial reality with force-feedback: Development of desktop virtual space with compact master manipulator. *ACM SIGGRAPH Comput Graphics* 24(4).

Iwata, H. 1990b. Artificial reality for walking about large-scale virtual space. *Human Interface News and Report* 5(1):49–52 (in Japanese).

Iwata, H. 1993. Pen-based haptic virtual environment. In *Proc of IEEE VRAIS '93*, 287–92. Seattle, WA: IEEE Computer Society.

Iwata, H. 1994. Desktop force display. In *SIGGRAPH '94 Visual Proceedings*.

Iwata, H. 1999. Walking about virtual space on an infinite floor. In *Proc. of IEEE Virtual Reality '99*, 286–93. Houston, Tx:IEEE.

Iwata, H., and T. Fujii. 1996. Virtual perambulator: A novel interface device for locomotion in virtual environment. *Proc. of IEEE 1996 Virtual Reality Annual International Symposium*, 60–5. Santa Clara, CA: IEEE Computer Society.

Iwata, H., and K. Matsuda. 1992. Haptic walkthrough simulator. In *Proc of ICAT '92*, 185–92.

Iwata, H., and Y. Yoshida. 1999. Path reproduction tests using a Torus Treadmill. *Presence* 8(6):587–97.

Kajimoto, H., N. Kawakami, T. Maeda, and S. Tachi. 1999. Tactile feeling display using functional electrical stimulation. In *Proc. of ICAT '99*, 107–14.

Kawai, Y., and F. Tomita. 2000. A support system for the visually impaired to recognize three-dimensional objects, IOS Press. *Technol Disability* 12(1):13–20.

Kontarinis, D. A., and R. D. Howe. 1995. Tactile display of vibratory information in teleoperation and virtual environment. *Presence* 4(4):387–402.

Lederman, S. J., and R. L. Klatzky. 1987. Hand movements: A window into haptic object recognition. *Cogn Psychol* 19(3):342–68.

Lederman, S. J., and R. L. Klatzky. 1999. Sensing and displaying spatially distributed fingertip forces in haptic interfaces for teleoperators and virtual environment system. *Presence* 8(1):86–103.

Lorenzo, M., et al. 1995. OSIRIS. In *SIGGRAPH '95 Visual Proceedings*, 129.

Massie, T., and K. Salisbury. 1994. The PHANToM haptic interface: A device for probing virtual objects. In *ASME Winter Annual Meeting*, DSC, Vol. 55–1, 295–9. American Society of Mechanical Engineers.

McNeely, W. 1993. Robotic graphics: A new approach to force feedback for virtual reality. In *Proc. of IEEE VRAIS '93*, 336–41. Seattle, WA: IEEE Computer Society.

Minsky, M., and S. J. Lederman. 1997. Simulated haptic textures: Roughness. Symposium on haptic interfaces for virtual environment and teleoperator systems. In *Proceedings of the ASME Dynamic Systems and Control Division*, DSC-Vol. 58, 421–6.

Moy, G., C. Wagner, and R. S. Fearing. 2000. A compliant tactile display for teletaction. In *IEEE Int Conf on Robotics and Automation*, 3409–15. San Francisco, CA: IEEE.

Murakami, T., and N. Nakajima. 1994. Direct and intuitive input device for 3D shape deformation. *ACM CHI 1994, Conference on Human Factors in Computing Systems*, 465–70. New York: ACM.

Noma, H., T. Sugihara, and T. Miyasato. 2000. Development of ground surface simulator for Tel-E-Merge system. In *Proceedings of IEEE Virtual Reality 2000*, 217–24. Washington, DC: IEEE Computer Society.

Poston, R., et al. 1997. A whole body kinematic display for virtual reality applications. In *Proc. of the IEEE International Conference on Robotics and Automation*, 3006–11.

Prat, D. R., et al. 1994. Insertion of an articulated human into a networked virtual environment. In *Proceedings of the 1994 AI, Simulation, and Planning in High Autonomy Systems Conference*, 84–90. Gainesville, FL: IEEE.

Sinclair, M. 1997. The haptic lens. In *SIGGRAPH '97 Visual Proceedings*, 179. New York: ACM.

Slater, M., M. Usoh, and A. Steed. 1994. Steps and ladders in virtual reality. In *Virtual Reality Technology*, 45–54. River Edge, NJ: World Scientific Publication.

Tachi, S., T. Maeda, R. Hirata, and H. Hoshino. 1994. A construction method of virtual haptic space. In *Proc. of ICAT '94*, 131–8. Tokyo, Japan.

Yamao, T., S. Ishida, S. Ota, and F. Kaneko. 1996. Formal safety assessment—research project on quantification of risk on lives. In *MSC67/INF.9 IMO Information Paper*.

10 Nonspeech Auditory and Crossmodal Output

Eve Hoggan and Stephen Brewster

CONTENTS

10.1 INTRODUCTION AND A BRIEF HISTORY OF NONSPEECH SOUND AND TACTILE FEEDBACK IN HUMAN–COMPUTER INTERACTION

Our senses of hearing and touch are very powerful and can convey a wealth of information. Sound gives us a continuous, holistic contact with our environment and what is going on around us. Nonspeech sounds (such as music, environmental sounds, or sound effects) give us different types of information to those provided by speech; they can be more general and ambient where speech is precise and requires more focus. Nonspeech sounds complement speech in the same way as visual icons complement text. For example, icons can present information in a small amount of space compared with text, whereas nonspeech sounds can present information in a small amount of time compared with speech.

Similarly, our sense of touch can also provide us with a vast amount of information. Geldard (1960) wrote: "for some kinds of messages the skin offers a valuable supplement to ears and eyes" (p. 1583). The skin offers a large display space, which can be used to display information (Geldard 1960). As the skin is often less engaged in other tasks than the eyes or ears, it is always ready to receive information (van Veen and van Erp 2000). Using the sense of touch enables subtle and private communication unlike sound.

The combination of visual, tactile, and auditory feedback at the user interface is a powerful tool for interaction. In our everyday life, these primary senses combine to give complementary information about the world. Blattner and Dannenberg (1992) discuss some of the advantages of using this approach in multimedia or multimodal computer systems: "In our interaction with the world around us, we use many senses. Through each sense, we interpret the external world using representations and organizations to accommodate that use. The senses enhance each other in various ways, adding synergies or further informational dimensions" (p. 5). These advantages can be brought to the multimodal (or crossmodal) human–computer interface by the addition of nonspeech auditory output with tactile feedback to standard graphical displays (see Chapter 18 for more on multimodal interaction). While directing our visual attention to one task, for example, while editing a document, we can still monitor the state of other tasks on our machine using sound and touch. Currently, almost all information presented by computers uses the visual sense. This means information can be missed because of visual overload or because the user is not looking in the right place at the right time. A multimodal interface that integrates information output to both senses could capitalize on the interdependence between them and present information in the most efficient way possible. An alternative approach is to use a crossmodal interface where the different senses are used to receive the same data. This provides a common representation of the data from both senses (in this case, audio and tactile) making them congruent informationally (Hoggan 2010a). Crossmodal use of the different senses allows the characteristics of one sensory modality to be transformed into stimuli for another sensory modality.

The classical uses of nonspeech sound can be found in human factors literature (see McCormick and Sanders 1982). Here, it is used mainly for alarms and warnings or monitoring and status information. Buxton (1989) extends these ideas and suggests that encoded messages could be used to present more complex information in sound, and it is this type of auditory feedback that will be considered here.

The use of sound to convey information in computers is not new. In the early days, programmers used to attach speakers to their computer's bus or program counter (Thimbleby 1990). The speaker would click each time the program counter was changed. Programmers would become accustomed to the patterns and rhythms of sound and could recognize what the machine was doing. Another everyday example is the sound of a hard disk. Users often can tell when a save or copy operation has completed by the noise the disk makes. This allows them to do other tasks while waiting for the copy to finish. Nonspeech sound is therefore an important information provider, giving users knowledge about factors they may not see.

Two important events kick-started the research area of nonspeech auditory output: the first was the special issue of the *Human-Computer Interaction* (HCI) journal in 1989 on nonspeech sound, edited by Bill Buxton (1989). This laid the foundations for some of the key works in the area; it included papers by Blattner on *earcons* (Blattner, Sumikawa, and Greenberg 1989), Gaver on *auditory icons* (Gaver 1989), and Edwards on *soundtrack* (Edwards 1989). The second event was the First International Conference on Auditory Display (ICAD'92) held in Santa Fe in 1992 (Kramer 1994b). For the first time, this meeting brought together the main researchers interested in the area (see www.icad.org for the proceedings of the ICAD conferences). Resulting from these two events was a large growth in research in the area during the 1990s that continues today.

Fifty years on from Geldard's original papers on the use of touch for information display (Geldard 1957), tactile feedback is now common in many everyday devices, with mobile phones, handheld computers, and game controllers all featuring vibration feedback. For example, vibrotactile actuators are often used in computer game controllers to provide feedback for weapon fire and environmental effects to provide a more immersive experience for the player.

Originally, work on vibrotactile displays was driven by tactile-audio substitution for persons with profound hearing impairment and was developed in the late 1970s and early 1980s. One of the earliest devices was Tacticon, a commercial device that adjusted the perceived intensity of 16 electrodes, each of which corresponded to a range of frequencies in the auditory spectrum, to improve speech comprehension, auditory discrimination, and the clarity of the users' speech (Kaczmarek and Bach-Y-Rita 1995).

There is commercial interest in this area too, as most mobile telephones include tactile feedback to accompany ring tones. For example, Immersion's VibeTonz (http://www.immersion.com) attempt to extend this simple feedback to enhance games and ring tones. Vibrotactile displays have been incorporated into canes used by visually impaired people. The UltraCane (http://www.soundforesight.co.uk) uses ultrasound to detect objects in a user's environment and presents the location and distance to targets by vibrating pads on the handle of the cane.

The combination of nonspeech audio and tactile feedback in computer interfaces can exploit the complementary relationship between the modalities and has the potential to increase the bandwidth through which information may be presented.

The rest of this chapter goes into detail on all aspects of auditory and crossmodal interface design. Section 10.2 presents some of the advantages and disadvantages of using sound at the interface. Then, Section 10.3 gives a brief introduction to psychoacoustics and psychophysics, or the study of the perception of sound and touch. Section 10.4 gives information about the basic sampling and synthesis techniques needed for auditory interfaces and Section 10.5 describes tactile actuator technology followed by tactile feedback generation techniques. Section 10.6 outlines the main techniques used for auditory and crossmodal information presentation, and Section 10.7 then goes through some of the main applications of sound and touch in HCI. The chapter finishes with some conclusions about the state of research in this area.

10.2 WHY USE NONSPEECH FEEDBACK IN HUMAN–COMPUTER INTERFACES?

10.2.1 Advantages of Nonspeech Feedback

There are many reasons why it is advantageous to use sound and touch at the user interface:

Vision and hearing are interdependent: Our visual and auditory systems work well together. Our eyes provide high-resolution information around a small area of focus (with peripheral vision extending further). On the other hand, sounds can be heard from all around the user: above, below, in front, or behind, but with a much lower resolution. Therefore, "our ears tell our eyes where to look": If there is an interesting sound from outside our view, we will turn to look at it to get more detailed information.

Hearing and touch have amodal properties: Our senses of hearing and touch share several important similarities, in particular their temporal characteristics and their ability to perceive vibrations. Moreover, sounds are often described in tactile terms. Mursell (1937) observed that tones can contain tactile values as can be seen when we describe a tone as hard or soft, rough or smooth, and wooden or metallic. An attribute that can communicate comparable information across modalities is considered to be amodal. Mendelson (1979) provided a scheme or list of such amodal properties. These properties relate to space and time and involve points along a continuum (e.g., location), intervals within continuum (e.g., duration), patterns of intervals (e.g., rhythm), rates of patterns (e.g., tempo), or changes of rate (e.g., texture gradients).

Sound has superior temporal resolution: As Kramer (1994a) says: "Acute temporal resolution is one of the greatest strengths of the auditory system." In certain cases, reactions to auditory stimuli have been shown to be faster than reactions to visual stimuli (Bly 1982).

Sound and touch reduce the overload from large displays: Modern, large, or multiple monitor graphical interfaces use the human visual sense very intensively. This means that we may miss important information because our visual system is overloaded—we have just too much to look at. To stop this overload, information could be displayed in sound or touch so that the load could be shared between senses.

Sound and touch reduce the amount of information needed on screen: Related to the above point is the problem with information presentation on devices with small visual displays, such as mobile telephones or personal digital assistants (PDAs). These have very small screens that can easily become cluttered. To solve this, some information could be presented in sound or touch to release screen space.

Sound reduces demands on visual attention: Another issue with mobile devices is that users who are using them on the move cannot devote all of their visual attention to the device—they must look where they are going to avoid uneven surfaces, traffic, pedestrians, and so on. In this case, visual information may be missed because the user is not looking at the device. If this were played in sound, then the information would be delivered while the user was looking at something else.

Sound is attention grabbing: Users can choose not to look at something, but it is harder to avoid hearing it. This makes sound very useful for delivering important information.

Touch is subtle and private: Using tactile feedback can enable personal and concealed communication, as it can only be felt by the user. Tactile feedback can, therefore, be used in situations where auditory display would be inappropriate (Chang et al. 2002), for example in meetings, or while in a quiet environment such as a library.

Spatial resolution of tactile stimuli is high: Different body locations have different levels of sensitivity and spatial acuity. The most sensitive part of the human body is the fingertip. When applying tactile stimuli to multiple points on the body, the distance between points is extremely important. Two-point discrimination is a measure that represents how far apart two pressure points must be before they are perceived as two distinct points on the skin (Geldard 1960). The point of contact discrimination threshold for two points is 0.9 mm when the stimuli are placed against the subject's finger in the absence of any movement lateral to the skin's surface. It is not possible for two points of contact closer than this threshold to be distinguished as separate stimuli. Experimental evidence suggests "active exploration marginally increases sensitivity, decreasing the threshold to 0.7 mm" (Phillips and Johnson 1985).

Auditory or tactile form makes computers more usable by visually disabled people: With the development of graphical displays, user interfaces became much harder for visually impaired people to operate. A screen reader (see Section 10.2.2) cannot easily read this kind of graphical information. Providing information in an auditory or tactile form can allow visually disabled persons to use the facilities available on modern computers.

10.2.2 Disadvantages of Nonspeech Feedback

Kramer (1994a) suggests some general difficulties with using sound to present information as follows:

Sound has low resolution: Many auditory and tactile parameters are not suitable for high-resolution display of quantitative information. Using sound volume or tactile amplitude, for example, only a very few different values can be unambiguously presented (Buxton, Gaver, and Bly 1991; Geldard 1960). The same also applies to spatial precision in sound, unlike touch. Under optimal conditions, differences of about 1° can be detected in front of a listener (see Section 10.4 and Blauert 1997). In vision, differences of an angle of 2 seconds can be detected in the area of greatest acuity in the central visual field.

Presenting absolute data is difficult: Many interfaces that use nonspeech feedback to present data do it in a relative way. Users hear or feel the difference between two sounds or vibrations to tell if a value is going up or down but absolute values are difficult.

There is lack of orthogonality: Changing one attribute of a sound or tactile cue may affect the others. For example, changing the frequency of stimuli may affect its perceived amplitude and vice versa (see Section 10.3).

There is annoyance due to auditory feedback: There are two aspects to annoyance: A sound may be annoying to the user whose machine is making the noise (the primary user) and/or annoying to others in the same environment who overhear it (secondary users). Buxton (1989) has discussed some of the problems of sound and suggests that some sounds help us (information) and some impede us (noise). We therefore need to design sounds so that there are more informative ones and less noise.

There are many studies of annoyance from speech (e.g., Berglund, Harder, and Preis 1994), from the sounds of aircraft, traffic, or other environmental noise and most of these suggest that the primary reason for the annoyance of sound is excessive volume. In a different context, Patterson (1989) investigated some of the problems with auditory warnings in aircraft cockpits. Many of the warnings were added in a “better safe than sorry” manner that lead to them being so loud that the pilot’s first response was to try and turn them off rather than deal with the problem being indicated.

A loud sound grabs the attention of the primary user, even when the sound is communicating an unimportant event. As the sound is loud, it travels from one machine to the ears of other people working nearby, increasing the noise in their environment.

So, how can annoyance be avoided? One key way is to avoid using intensity as a cue in sound design for auditory interfaces. Quiet sounds are less annoying. Listeners are also not good at making absolute intensity judgments (see Section 10.3). Therefore, intensity is not a good cue for differentiating sounds anyway.

Manipulating sound parameters other than intensity can make sounds attention grabbing (but not annoying). Rhythm or pitch can be used to make sounds demanding because the human auditory system is very good at detecting changing stimuli (for more see Edworthy, Loxley, and Dennis [1991]). Therefore, if care is taken with the design of sounds in an interface, specifically avoiding the use of volume changes to cue the user, then many of the problems of annoyance can be avoided.

The other key way to avoid annoyance is to use the tactile modality instead. At some times, audio feedback may be appropriate and other times the tactile modality may be more appropriate, for example, when surrounded by others. A system could switch to tactile cues in these circumstances (see Section 10.6.5).

10.3 BRIEF INTRODUCTION TO AUDITORY AND TACTUAL PERCEPTION

10.3.1 Auditory Perception

This section provides some basic information about the perception of sound that is applicable to nonspeech auditory output. Auditory interface designers must be conscious of the effects of psychoacoustics, or the perception of sound, when designing sounds for an interface. As Frysinger (1990) says: “The characterization of human hearing is essential to auditory data representation because it defines the limits within which auditory display designs must operate if they are to be effective. There is not enough space here to give great detail on this complex area, for more see the study by Moore (2003).

What is sound? Sounds are pressure variations that propagate in an elastic medium (in this case, the air). The pressure variations originate from the motion or vibration of objects. These pressure variations hit the listener’s ear and start the process of sound perception. A sine wave could be considered the simplest form of sound (as might be produced by a tuning fork). A sound is made up of three basic components:

Frequency is the number of times per second the wave repeats itself (Figure 10.1 shows three cycles). It is normally measured in hertz (Hz). *Amplitude* is the deviation away from the mean pressure level, or force per unit area of a sound. It is normally measured in decibels (dB). *Phase* is the position of the start of the wave on the time axis (measured in milliseconds).

Sounds from the real world are normally much more complex than that shown in Figure 10.1. and tend to comprise many sine waves with different frequencies, amplitudes, and phases. Figure 10.1 shows a more complex sound made of three sine wave components (or *partials*) and the resulting waveform. *Fourier analysis* allows a sound to be broken down into its component sine waves (Gelfand 1981).

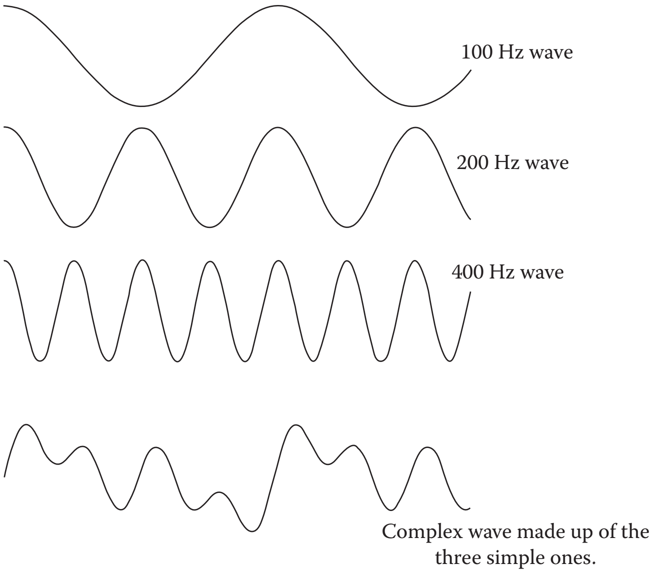


FIGURE 10.1 A complex wave made up of three components with its fundamental at 100 Hz.

The sounds in Figures 10.1 are *periodic*—they repeat regularly over time. This is very common for many types of musical instruments that might be used in an auditory interface. Many natural, everyday sounds (such as impact sounds) are not periodic and do not repeat. The sounds in Figure 10.1 are also *harmonic*—their partials are integer multiples of the lowest (or *fundamental*) frequency. This is again common for musical instruments but not for everyday sounds. Periodic harmonic sounds have a recognizable pitch, whereas nonperiodic, inharmonic sounds tend to have no clear pitch.

The attributes of sound described above are the physical aspects. There is a corresponding set of perceptual attributes. *Pitch* is the perceived frequency of a sound. Pitch is roughly a logarithmic function of frequency. It can be defined as the attribute of auditory sensation in terms of which sounds may be ordered on a musical scale (Moore 2003). In Western musical systems, there are 96 different pitches arranged into eight octaves of 12 notes. Tones separated by an octave have the frequency ratio 2:1. For example, middle C is 261.63 Hz, the octave above this is at 523.25 Hz, and the octave below at 130.81 Hz. It is one of the most useful and easily controlled aspects of sound and is very useful for auditory interface designers. However, as Buxton, Gaver, and Bly (1991) say: “It is important to be aware of the myriad interactions between pitch and other attributes of sound ….” For example, pitch is affected by sound intensity: at less than 2 kHz an increase in intensity increases the perceived pitch, and at 3 kHz and above an increase in intensity decreases the perceived pitch (Gelfand 1981).

Humans can perceive a wide range of frequencies. The maximum range we can hear is from 20 Hz to 20 kHz. This decreases with age so that at the age of 70 a listener might only hear a maximum of 10 kHz. It is therefore important to ensure that the sounds in an auditory interface are perceivable by its users (also poor-quality loudspeakers may not be able to cope with the highest or lowest frequencies). Listeners are not good at making absolute judgments of pitch (Moore 2003). Only 1% of the population has *perfect pitch*. Another important factor is *tone deafness*. Moore suggests that this is a misnomer and almost everyone is able to tell that two sounds are different; they are not always able to say which is higher or lower. Mansur, Blattner, and Joy (1985) give evidence of one other important effect: “There appears to be a natural tendency, even in infants, to perceive a pitch that is higher in frequency to be coming from a source that is vertically higher in space when compared to some lower tone.” This is important when creating an auditory interface as it could be used to give objects a spatial position. If only stereo position is available to provide spatial cues in the horizontal plane, then pitch could provide them in the vertical plane. Guidelines for the use of pitch (and the other parameters below) are described in Section 10.6.

Loudness is the perceived intensity of a sound. Loudness (*L*) is related to intensity (*I*) according to the *Power Law*: $L = k\,I^{0.3}$ (Gelfand 1981). Therefore, a 10-dB increase in intensity doubles the perceived loudness of a sound. Loudness is again affected by the other parameters of sound. For example, sounds of between 1 and 5 kHz sound louder at the same intensity level than those outside that frequency range. Humans can perceive a very wide range of intensities: the most intense sound that a listener can hear is 120 dB louder than the quietest. This equates to a ratio of 1,000,000,000,000:1 (Moore 2003). Buxton, Gaver, and Bly (1991) also report that listeners are “very bad at making absolute judgments about loudness” and “our ability to make relative judgments of loudness are limited to a scale of about three different levels.” It is also a primary cause of annoyance (see Section 10.2) so should be used sparingly by auditory interface designers.

Timbre is the “quality” of the sound. It is the attribute of auditory sensation in terms of which a listener can judge two sounds with the same loudness and pitch to be dissimilar. It is what makes a violin sound different to a piano even if both are playing the same pitch at the same loudness. Its structure and dimensions are not yet fully understood. It is known to be based partly on the spectrum and dynamics of a sound. As Blattner, Sumikawa, and Greenberg (1989) say: “Even though timbre is difficult to describe and notate precisely, it is one of the most immediate and easily recognizable characteristics of sound” (both *auditory icons* and *earcons* use timbre as one of their fundamental attributes—see Section 10.6).

Duration is another important attribute of sound. Sounds of different durations are used to form rhythmic structures that are a fundamental part of music. Duration can also affect the other parameters of sound. For example, for sounds of less than 1 second, loudness increases with duration. This is important in auditory interfaces because short sounds are often needed so that the auditory feedback can keep pace with the interactions taking place, accordingly, they must be made loud enough for listeners to hear.

Direction is the position of the sound source. If a sound source is located to one side of the head, then the sound reaching the further ear will be reduced in intensity (*interaural intensity difference*—IID) and delayed in time (*interaural *time difference*—ITD) (Blauert 1997). These are two key factors allowing a listener to *localize* a sound in space. Humans can detect small changes in the position of a sound source. The minimum auditory angle is the smallest separation between two sources that can be reliably detected. Strybel, Manligas, and Perrott (1992) report that in the median plane sound sources only 1° apart can be detected. At 90° azimuth (directly opposite one ear), accuracy falls to ± 10° (see Section 10.4.3 for more on sound positioning).

10.3.2 Tactual Perception

The term *haptics* means "sensory and/or motor activity based in the skin, muscles, joints and tendons" (*ISO: Ergonomics of human-computer interaction—Part 910: Framework for tactile and haptic interaction* 2009). Under this umbrella term, however, there are several subcategories, as shown in Figure 10.2.

The skin has an area of 1.8 m^2, a density of 1250 kg/m^3, and a weight of 5 kg (Sherrick and Cholewiak 1986). It is classified as either glabrous (i.e., nonhairy) skin, which is found only on the plantar and palmar surfaces, or hairy skin, which is found on the rest of the body. These divisions are relevant to tactile displays because they vary in sensory receptor systems and measures of tactile sensitivity (Cholewiak and Craig 1984). Four types of mechanoreceptive fibers have been identified in glabrous skin: Meissner corpuscle (RA), Merkel cell (SAI), Pacinian corpuscle (PC), and Ruffini ending.

Each mechanoreceptive fiber has a specific role in the perception of vibration that ranges from 0.4 to more than 500 Hz (Sherrick and Cholewiak 1986). The RA are high-density fibers that are abundant in the fingertips. The majority of the tactile feedback used in this thesis research is presented to the fingertips. In contrast, the PCs are less dense than the RA, and are numerous in the distal joints. Since the four fibers overlap in their absolute sensitivities, a vibration stimulus will seldom stimulate a single fiber in the skin but several fibers because the energy applied to the skin will move throughout nearby skin tissues (Sherrick and Cholewiak 1986). Within most of the vibrotactile literature, the fibers are grouped into two systems: the Pacinian and the non-Pacinian systems. "The Pacinian system has a large receptive field excited by higher frequencies and the non-Pacinian system consists of a small receptive field thought to be excited by lower frequencies" (Sherrick, Cholewiak, and Collins 1990). Bolanowski et al. (1988) found threshold sensitivities in the range of 0.4–500 Hz between these two systems. The Pacinian system exhibited a U-shaped function at higher frequencies (40–500 Hz) where maximum sensitivity occurred between 250 and 300 Hz (Bolanowski et al. 1988). Therefore, the majority of the stimuli used in this research have a frequency of 250 Hz. Verrillo (1966) also reported a similar function for hairy skin, where maximum sensitivity occurred at 220 Hz.

Understanding the features of specific skin fibers and their response characteristics when stimulated can help to inform the design of any tactile feedback to ensure that the stimuli are compatible with the characteristics of the particular area of skin on which the feedback will be presented. According to Kandel and Jessell (1991), Meissner's corpuscles and Merkel's cells respond to touch, PCs respond to vibration, and Ruffini's corpuscles respond to rapid indentation of the skin. Vibration is detected best on hairy, bony skin and is more difficult to detect on soft, fleshy areas of the body (Geldard 1960).

The dimensions or attributes of our sense of touch are detailed below:

Frequency range of the skin ranges from 10 to 400 Hz, with maximum sensitivity (Summers, Dixon, and Cooper 1994) and finer spatial discrimination at around 250 Hz (Craig and Sherrick 1982). Investigations by Goff involving the stimulation of the subject's finger with a single probe showed that for lower frequencies (<25 Hz), the discrimination threshold was less than 5 Hz. For frequencies greater than 320 Hz, discrimination capacities were also degraded (Goff 1967). Measures for discrimination thresholds of frequency are problematic, as perception of vibratory pitch is

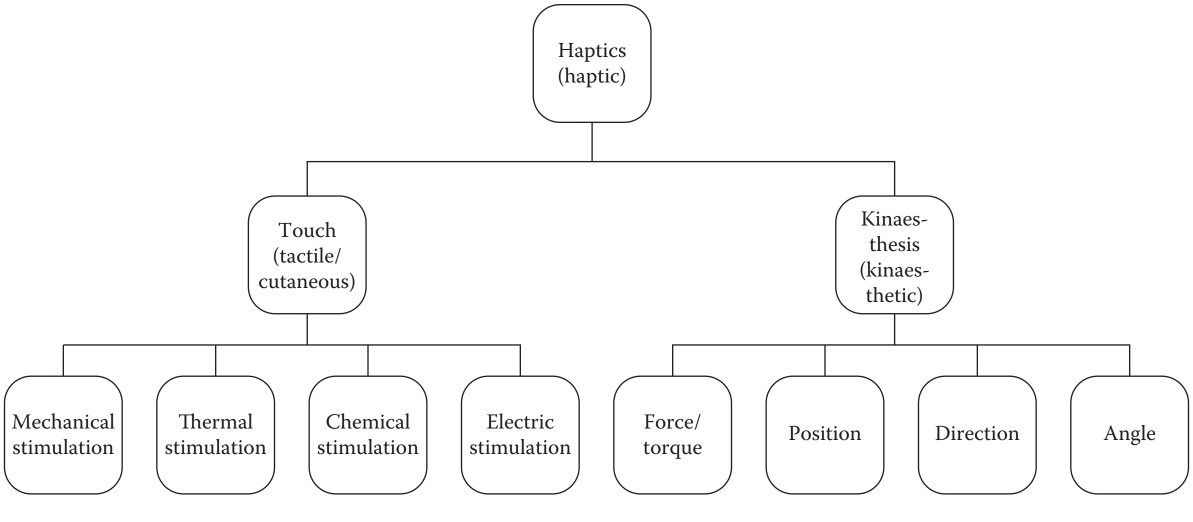


FIGURE 10.2 Definitions of terminology. (Adapted from ISO: Ergonomics of human-computer interaction—Part 910: Framework for tactile and haptic interaction 2009.)

dependent not just on frequency, but also on the amplitude of stimulation. Geldard (1957) found that subjects reported a change in pitch when frequency was fixed, but amplitude of stimulation was changed. Sherrick (Sherrick and Cholewiak 1986) found that combining frequency and amplitude redundantly allowed a greater number of identifiable levels to be created. He found that people could distinguish three to five different levels of frequency, but that adding amplitude as a redundant parameter could increase this range. Therefore, this interaction between frequency and amplitude should to be taken into account or perhaps avoided when designing tactile stimuli.

Duration is another important dimension found in both the audio and tactile modalities. Geldard (1960) reports that the temporal duration just noticeable difference rose from 50 to 150 milliseconds. When duration was increased from 0.1 to 2.0 seconds Gescheider (as reported in [Terhardt 1974]) measured the time difference between two tactile "clicks" on the fingertip, necessary for them to be perceived as two separate sensations and found that the minimum threshold reported was 10 milliseconds. Interactions between duration and perceived amplitude should be considered when using duration, as it has been shown that short intense signals can be confused with longer, lower intensity signals. Gunther (Gunther, Davenport, and O'Modhrain 2002) suggests that stimuli lasting less than 0.1 seconds may be perceived as taps or jabs, whereas longer stimuli may be perceived as smoothly flowing tactile phrases. Craig and Sherrick (1982) warn that very short durations may result in sensations such as pokes or jabs, which might be undesirable.

Rhythms are created by grouping together pulses to create temporal patterns in a similar fashion to rhythms in music. Rhythm is very important and useful in the design of tactile systems. For example, Summers (1992) encoded speech information by modulating vibration frequency and amplitude, and by presenting the temporal pattern of the speech using rhythm. The results of an evaluation showed that users obtained the most information from the rhythmic pattern compared with the frequency/amplitude modulation.

Body locations that have been identified as most sensitive to pressure and stimulus discrimination include the finger, hand, arm, thigh, and torso (Cholewiak and Collins 1995).

Cholewiak, Brill, and Schwab (2004) investigated the vibrotactile localization accuracy for the abdomen using 12, 8, and 6 equidistant actuators, 72, 107, and 140 mm, respectively. Their results showed that the ability to correctly identify which actuator was presenting a stimulus increased as the number of actuators decreased. Study participants were correct in their identification for an average of 74%, 92%, and 97% of the trials for 12, 8, and 6 actuators, respectively. The results also showed that when participants labeled areas on their abdomen, for example, the navel at 12 o'clock and the spine at 6 o'clock, they were better able to localize stimuli. Accuracy rates were much lower when labels were not available. This suggests that accuracy can be increased if a label is provided which is mapped to the locations to be identified.

Intensity detection in our sense of touch is limited, with an intensity range of approximately 55 dB above the detection threshold. Any vibrations above this threshold feel unpleasant or even painful (Verrillo and Gescheider 1992).

10.4 TECHNOLOGY AND PRODUCTION OF NONSPEECH AUDIO FEEDBACK

Most desktop PCs, handheld computers, and mobile telephones have sophisticated sound hardware available for auditory interface designers. This is normally for playing games but is sufficient to do most of the things required by an auditory interface. The aim of this section is to describe briefly some of the main technologies that are important for the designers to understand when creating interfaces.

There are two main aspects of sound production: (1) sound synthesis and (2) sound sampling and playback. A basic overview focusing on aspects related to audio interfaces will be given. For much more detail on sound synthesis and musical instrument digital interface (MIDI), see the study by Roads (1996) and Miranda (1998). For more on sampling, see the study by Pohlmann (2005).

There are many tools available for synthesis and sample playback, and devices from desktop PC's to mobile telephones have the processing power necessary. The Java programming language (www.java.sun.com), for example, has built-in support for a range of synthesis and sampling techniques for many different platforms. Libraries such as Fmod (www.fmod.org) allow standard crossplatform sound and work on many different devices and programming languages. All current desktop operating systems provide support for synthesis and sampling. The basic technologies necessary to make the sounds needed for auditory interfaces are thus readily available, but it is important to know something of how they work to use them most effectively.

10.4.1 Brief Introduction to Sound Synthesis and MIDI

The aim of sound synthesis is to generate a sound from a stored model, often a model of a musical instrument. For auditory interfaces, we need a wide and good quality range of sounds that we can generate in real time as the user interacts with the interface. Synthesizers come in three main forms: (1) soundcards on PCs, (2) external hardware devices, and (3) software synthesizers. The main forms of synthesis will now be briefly reviewed.

Wavetable synthesis is one of the most common and low-cost synthesis techniques. Many of the most popular PC soundcards use this technique (such as the SoundBlaster series from Creative Technology—www.creative.com). The idea behind wavetable synthesis is to use existing sound recordings (which are often very difficult to synthesize exactly) as the starting point and create very convincing simulations of acoustical instruments based on them (Miranda 1998; Roads 1996). A sample (recording) of a particular sound will be stored in the soundcard. It can then be played back to produce a sound. The sample memory in these systems contains a large number of sampled sound segments and can be thought of as a "table" of sound waveforms that may be looked up and used when needed. Wavetable synthesizers use a variety of different techniques, such as sample looping, pitch shifting, mathematical interpolation, and polyphonic digital filtering, to reduce the amount of memory required to store the sound samples and to get more types of sounds. More sophisticated synthesizers contain more wavetables (perhaps one or more for the initial *attack* part of a sound and then more for the *sustain* part of the sound and then more for the final *decay* and *release* parts). It is also possible to combine multiple separately controlled wavetables to create a new instrument.

Wavetable synthesis is not so good if you want to create new timbres as it lacks some of the flexibility that other techniques have. Most wavetable synthesizers contain many sounds (often many hundreds) so there may not be a great need to create new ones. For most auditory interfaces, the sounds from a good quality wavetable synthesizer will be perfectly acceptable. For desktop computers, the storage of large wavetables is no problem, but for mobile telephones with less storage, there may be a much smaller, lower quality set, so care may be needed to design appropriate sounds.

Frequency modulation (FM) synthesis techniques generally use one periodic signal (the modulator) to modulate the frequency of another signal (the carrier) (Chowning 1975). If the modulating signal is in the audible range, then the result will be a significant change in the timbre of the carrier signal. Each FM voice requires a minimum of two signal generators. Sophisticated FM systems may use four or six operators per voice, and the operators may have adjustable envelopes that allow adjustment of the attack and decay rates of the signal. FM synthesis is cheap and easy to implement and can be useful for creating expressive new synthesized sounds. However, if the goal is to re-create the sound of an existing instrument, then FM synthesis is not the best choice as it can generally be done more easily and accurately with wavetable-based techniques.

Additive (and subtractive) synthesis is the oldest form of synthesis (Roads 1996). Multiple sine waves are added together to produce a more complex output sound (subtractive synthesis is the opposite: a complex sound has frequencies filtered out to create the sound required). Using this method, it is theoretically possible to create any sound (as all complex sounds can be decomposed into sets of sine waves by Fourier analysis). However, it can be very difficult to create any particular sound.

Physical modeling synthesis uses mathematical models of the physical acoustic properties of instruments and objects. Equations describe the mechanical and acoustic behavior of an instrument. The better the simulation of the instrument the more realistic the sound produced. Nonexistent instruments can also be modeled and made to produce sounds. Physical modeling is an extremely good choice for synthesis of many classical instruments, especially those of the woodwind and brass families. The downside is that it can require large amounts of processing power, which limits the polyphony.

10.4.1.1 Musical Instrument Digital Interface

The *MIDI* allows the real-time control of electronic instruments (such as synthesizers, samplers, drum machines) and is now very widely used (www.midi.org). It specifies a hardware interconnection scheme, a method for data communications and a grammar for encoding musical performance information (Roads 1996). For auditory interface designers, the most important part of MIDI is the performance data, which is a very efficient method of representing sounds.

MIDI performance information is like a piano roll: notes are set to turn on or off and play different instruments over time. A MIDI message is an instruction that controls some aspect of the performance of an instrument. The MIDI message is made up of a *status byte*, which indicates the type of the message, followed by up to two *data bytes*, which give the parameters. For example, the *Note On* command takes two parameters: one value giving the pitch of the note required and the other the volume. This makes it a very compact form of presentation.

Performance data can be created dynamically from program code or by a sequencer. In an auditory interface, the designer might assign a particular note to a particular interface event—for example, a click on a button. When the user clicks on the button, a MIDI Note On event will be fired, and when the user releases the button, the corresponding Note Off event will be sent. This is a very simple and straightforward way of adding sounds. With a sequencer, data can be entered using classical music notation by dragging and dropping notes onto a stave or using an external piano-style keyboard. This could then be saved to a MIDI file for later playback (or could be recorded and played back as a sample—see Section 10.4.2).

10.4.2 Brief Introduction to Sampling

In many ways, sampling is simpler than synthesis. The aim is to make a digital recording of an analog sound and then to be able to play it back later, with the played back sound matching the original as closely as possible. There are two important aspects: *sample rate* and *sample size*.

10.4.2.1 Sample Rate

This is the number of discrete "snapshots" of the sound that are taken, often measured per second. The higher the sampling rate, the higher the quality of the sound when it is played back. With a low sampling rate, few snapshots of the sound are taken and the recording will not match well the sound being recorded. The *sampling theorem* (Roads 1996) states that "In order to be able to reconstruct a signal, the sampling frequency must be at least twice the frequency of the signal being sampled" (p. 30). The limit of human hearing is about 20 kHz; therefore, a maximum rate of 40 kHz is required to be able to record any sound that a human can hear. The standard audio CD format uses a sample rate of 44.1 kHz, meaning that it covers all of the frequencies that a human can hear. If a lower sampling rate is used, then higher frequencies are lost. For example, the .au audio file format uses a sampling rate of 8 kHz, meaning that only frequencies of less than 4 kHz can be recorded. For more details on the huge range of sample formats, see the study by Bagwell (1998).

Higher sampling rates generate much more data than do lower ones so may not always be suitable if storage is limited (e.g., on a mobile computing device). Auditory interface designers should consider the frequency range of the sounds needed in an interface, and this might reduce the sample rate.

10.4.2.2 Sample Size

The larger the sample size the better the quality of the recording, as more information is stored at each snapshot of the sound. Sample size defines the volume (or dynamic) range of the sound. With an 8-bit sample, only 256 discrete amplitude (or quantization) levels can be represented. Fitting an analog sound into one of these levels might cause it to be rounded up or down, and this can add noise to the recording. CD-quality sounds use 16-bit samples giving 65,536 different levels, so the effects of quantization are reduced. Many high-quality samplers use 24-bit samples to reduce the problems of quantization still further.

The main two bit sizes used in most soundcards are 8 and 16 bits. As with sample rates, the main issue is size: 16-bit samples require a lot of storage, especially at high sample rates. Audio CD-quality sound generates about 10 megabytes of data per minute. Compression techniques such as MP3 can help reduce the amount of storage but keep quality high.

10.4.3 Three-Dimensional Sound

Much of the recorded sound we hear is in *stereo.* A stereo recording uses intensity differences between the ears of a listener. From these differences, the listener can gain a sense of movement and position of a sound source in the stereo field. The perceived position is along a line between two loudspeakers or inside the head between listeners' ears if they are wearing headphones. This simple, inexpensive technique can give useful spatial cues at the auditory interface. This is being taken further to make sounds appear as coming from around a user (in virtual 3D) when only a small number of loudspeakers (or even just a pair of headphones) are used. Spatial sound can be used for a range of things, including giving directional information, spreading sound sources around the head to help users differentiate simultaneous sounds, and to create "audio windows" in which to present information (see Section 10.7.3).

In the real world we use our pinnae (the outer ear) to filter the sounds coming from different directions so that we know where they are coming from. To simulate sounds as coming from around listeners and outside the head when wearing headphones, sounds entering the ear are recorded by placing microphones into the ear canals of listeners. The differences between the sound at the sound source and at the ear canal are then calculated, and the differences, or "head-related transfer functions" (HRTFs), derived are used to create filters with which stimuli can be synthesized (Blauert 1997). This research is important as 3D auditory interfaces can be created that are more natural, with sounds presented around the user as they would be in real life. Almost all current PC soundcards can generate such 3D sounds as they are often used in games.

The main problem with providing simulated 3D sound through headphones comes from the general HRTFs used. If listeners' ears are not like the ears of the head (often a dummy head) from which the HRTFs were generated, then the listeners are likely to feel that the sounds are coming from inside their head, and not outside. It is also very easy for listeners to become confused when they cannot tell whether a sound is in front or behind them. Vertical positioning is also difficult to do reliably. This means that many designers who use spatial sound in their interfaces often limit themselves to a plane cutting through the head horizontally at the level of the ears, creating a 2.5D space. This reduces the space in which sounds can be presented but avoids many of the problems of users not being able to localize the sounds properly.

To improve quality, head-tracking is often used. Once the orientation of the user's head is known, sounds can be respatialized to remain in position when the head turns. "Active listening" is used to disambiguate the location of a sound—listeners naturally make small head movements, and these change the IID and ITD, cueing the listeners to the location of the sound. Using such tracking can significantly improve the performance of 3D auditory user interfaces (Marentakis and Brewster 2004). Marentakis and Brewster also showed that targets should be around ±10° in size when head-tracking is used to enable accurate localization and selection.

10.5 TECHNOLOGY AND PRODUCTION OF TACTILE FEEDBACK

10.5.1 Tactile Actuators

Tactile devices generally stimulate the cutaneous senses through skin indentation, vibration, skin stretch, and electrical stimulation (Brewster et al. 2008). A variety of tactile stimulation devices are available, each of which stimulates a specific tactile response. These include pressure, thermal, slip, electrocutaneous, and vibration displays. In general, vibrotactile actuators are the easiest to work with and in particular, to control. Furthermore, most mobile devices already include a vibrotactile actuator.

Vibrotactile actuators can provide a sustained feedback allowing many different textures and intensities to be presented. By using the built-in actuator in commercial devices, the tactile feedback is not restricted by expensive or rare technology and does not require any hardware to be added to the device, which could increase its size, weight, or battery consumption, which may be inappropriate in mobile devices for example.

Most vibrotactile actuators stimulate the skin using electromagnetic actuation to drive a mass in either a linear or rotational manner. The Engineering Acoustics Inc (EAI) C2 Tactor is shown in Figure 10.3. This device is resonant

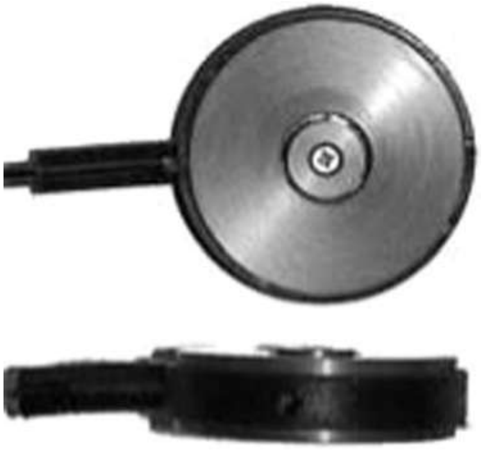


FIGURE 10.3 Engineering Acoustics Inc C2 vibrotactile actuator.

at 250 Hz with much reduced response at other frequencies (which is another reason for the reduced usefulness of frequency as a parameter for vibrotactile interfaces). The advantage of vibrotactile cues is that they can apply high levels of force (so can be felt through clothing) and they can also be distributed over the body or device to give spatial cues (often attached to a user's belt around the waist). For a more detailed review of vibrotactile devices, see the study by Summers (1992).

Piezoelectric actuators can create short, more display-localized tactile bursts by moving touch-screen display modules within the device (Laitinen and Mäenpää 2006). Piezoelectric actuators are also able to generate quick pulses, and the tactile feedback is concentrated to move the display mass, which is commonly 20% of the whole device mass, providing large displacement with rapid responses, but with less kinetic energy compared with traditional vibration motor systems.

Koskinen, Kaaresoja, and Laitinen (2008) conducted three laboratory-based studies to determine which tactile click (from a set of various different designs) is most pleasant to use in fingertip interaction with a mobile touch-screen device. Using two different types of actuator—piezo or vibration motor—the experiments allowed the authors to find the most pleasant tactile feedback as perceived by participants. The results showed that feedback from piezoelectric actuators was perceived as more pleasant than that from vibrotactile motors.

10.5.2 Tactile Feedback Generation

It is difficult to summarize tactile feedback generation as it depends heavily on the type of actuator used. Given that this chapter focuses mainly on vibrotactile feedback, this section outlines generation methods for the most commonly used vibrotactile actuators: voice coil motors (e.g., the C2 Tactor) and eccentric rotating mass (ERM) motors (e.g., standard built-in mobile phone motors).

As mentioned in Section 10.3, the dimensions or parameters of touch include amplitude, frequency, and rhythm. All these can be controlled in tactile feedback generation.

10.5.2.1 Using Audio Synthesis or Sampling Software for External Voice Coil Motors

Much like audio, tactile feedback can be generated using synthesis or sampling methods. Audio is played back normally from an application but instead of using speakers, audio output is forwarded to the actuators. Sampling behaves in the same way as audio sampling. In synthesis applications, both frequency and amplitude can be controlled somewhat independently, but it must be noted that actuators always have an optimum frequency. The human fingertip is most sensitive to a frequency of 250 Hz, and actuators such as the C2 are best used at this frequency. Most actuators have an optimum frequency in the range of 100–300 Hz. Virtually, any waveform can be used but most actuators are optimized for sine waves. Duration can also be easily controlled, meaning that rhythms can be created (as discussed in Section 10.3, rhythm is one of the most important and effective design parameters). Most standard commercial audio synthesis programs can be used such as Pure Data (http://puredata.info).

10.5.2.2 Using APIs and Eccentric Rotating Mass

When using ERMs (found in mobile phones), there are two issues to consider. First, it takes time to start and stop the rotating mass, which means there may be a slight delay in feedback. Second, when the rotating mass is mounted inside a device, the vibration disperses and can be felt across the whole device. Using standard application programming interfaces (APIs) provided by mobile phone manufacturers, for example, QT or in JavaME, it is possible to manipulate several parameters. It is possible to change the frequency by adjusting the supply voltage provided to the rotating mass. The direction of rotation can also be changed; alternating between two opposite driving directions can provide very distinctive feedback. Rhythms can be created by creating structured units of pulses and pauses.

10.6 NONSPEECH FEEDBACK TECHNIQUES

The two main types of nonspeech audio presentation techniques commonly used are *auditory icons* and *earcons*. There are also two main types of tactile presentation techniques called haptic icons and tactons. Following on from the audio and tactile presentation techniques, crossmodal icons can be used to make effective use of both modalities. Substantial research has gone into developing all of these, and the main work is reviewed in Section 10.6.1.

10.6.1 Auditory Icons

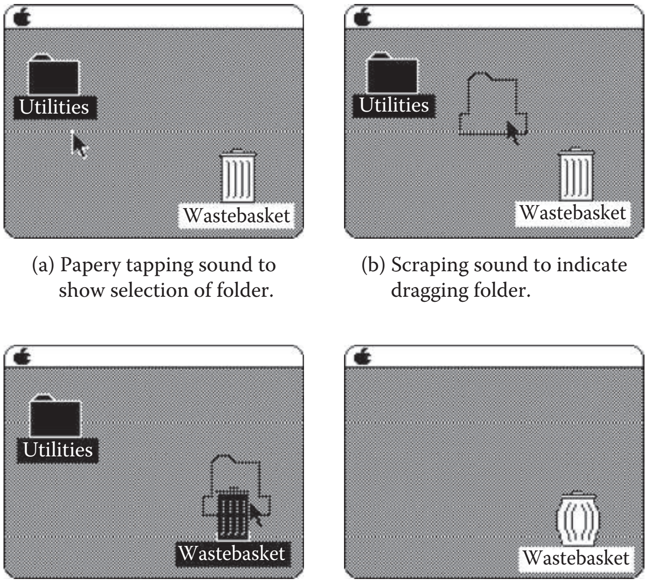
Gaver (1989, 1997) developed the idea of *auditory icons*. These are natural, everyday sounds that can be used to represent actions and objects within an interface. He defined them as "everyday sounds mapped to computer events by analogy with everyday sound-producing events. Auditory icons are like sound effects for computers." Auditory icons rely on an analogy between the everyday world and the model world of the computer (Gaver 1997) (for more examples of the use of earcons, see the work on Mercator and Audio Aura described in Sections 10.7.2 and 10.7.3).

Gaver used sounds of events that are recorded from the natural environment, for example, tapping or smashing sounds. He used an "ecological listening" approach (Neuhoff 2004), suggesting that people do not listen to the pitch and timbre of sounds but to the sources that created them. When pouring liquid, a listener hears the fullness of the receptacle, not the increases in pitch. Another important property of everyday sounds is that they can convey multidimensional data. When a door slams, a listener may hear the following: the size and material of the door; the force that was used; and the size of room where the door was slammed. This could be used within an interface so that selection of an object makes a tapping sound, the type of material could represent the type of object, and the size of the tapped object could represent the size of the object within the interface.

Gaver used these ideas to create auditory icons and from these built the *SonicFinder* (Gaver 1989). This ran on the Apple Macintosh and provided auditory representations of some objects and actions within the interface. Files were given a wooden sound, applications a metal sound, and folders a paper sound. The larger the object the deeper the sound it made. Thus, selecting an application meant tapping it—it made a metal sound, which confirmed that it was an application and the deepness of the sound indicated its size. Copying used the idea of pouring liquid into a receptacle. The rising of the pitch indicated that the receptacle was getting fuller and the copy progressing.

To demonstrate how the SonicFinder worked, a simple interaction is provided in Figure 10.4, showing the deletion of a folder. In Figure 10.4a, a folder is selected by tapping on it; this causes a "papery" sound indicating that the target is a folder. In Figure 10.4b, the folder is dragged toward the wastebasket causing a scraping sound. In Figure 10.4c, the wastebasket becomes highlighted, and a "clinking" sound occurs when the pointer reaches it. Finally, in Figure 10.4d, the folder is dropped into the wastebasket and a smashing sound occurs to indicate it has been deleted (the wastebasket becomes "fat" to indicate there is something in it).

Problems can occur with representational systems such as auditory icons because some abstract interface actions and objects have no obvious representation in everyday sounds. Gaver used a pouring sound to indicate copying because there was no natural equivalent; this is more like a "sound effect." He suggested the use of movie-like sound effects to create sounds for things with no easy representation. This may cause problems if the sounds are not chosen correctly as they will become more abstract than representational, and the advantages of auditory icons will be lost.



(a) Papery tapping sound to show selection of folder.

(b) Scraping sound to indicate dragging folder.

(c) Clinking sound to show wastebasket selected.

(d) Smashing sound to indicate folder deleted.

FIGURE 10.4 An interaction showing the deletion of a folder in the SonicFinder. (From Gaver, W. 1989. *Hum Comput Interact* 4(1):67–94. With permission.)

Gaver developed the ideas from the SonicFinder further in the *ARKola* system (Gaver, Smith, and O'Shea 1991), which modeled a soft drinks factory. The simulation consisted of a set of nine machines split into two groups: those for input and those for output. The input machines supplied the raw materials; the output machines capped the bottles and sent them for shipping. Each machine had an on/off switch and a rate control. The aim of the simulation was to run the plant as efficiently as possible, avoid waste of raw materials, and make a profit by shipping the bottles. Two users controlled the factory, with each user able to see approximately one-third of the whole plant. This form of plant was chosen because it allowed Gaver et al. to investigate how the sounds would affect the way users handled the given task and how people collaborated. It was also an opportunity to investigate how different sounds would combine to form an auditory *ecology* (integrated set of sounds) or soundscape. Gaver et al. related the way the different sounds in the factory combined to the way a car engine is perceived. Although the sounds are generated by multiple distinct components, they combine to form what is perceived as a unified sound. If something goes wrong, the sound of the engine will change, alerting the listener to the problem, but in addition, to a trained ear the change in the sound would alert the listener to the nature of the problem. The sounds used to indicate the performance of the individual components of the factory were designed to reflect the semantics of the machine.

Each machine had a sound to indicate its status over time, for example, the bottle dispenser made the sound of clinking bottles. The rhythm of the sounds reflected the rate at which the machine was running. If a machine ran out of supplies or broke down, its sound stopped. Sounds were also added to indicate that materials were being wasted. A splashing sound indicated that liquid was being spilled; the sound of smashing bottles indicated that bottles were being lost. The system was designed so that up to 14 different sounds could be played at once. To reduce the chance that all sounds would be playing simultaneously, sounds were pulsed once a second rather than playing continuously.

An informal evaluation was undertaken where pairs of users were observed controlling the plant, either with or without sound. These observations indicated that the sounds were effective in informing the users about the state of the plant and that the users were able to differentiate the different sounds and identify the problem when something went wrong. When the sounds were used, there was much more collaboration between the two users. This was because each could hear the whole plant and therefore help if there were problems with machines that the other was controlling. In the visual only condition, the users were not as efficient at diagnosing what was wrong even if they knew there was a problem.

One of the biggest advantages of auditory icons is the ability to communicate meanings that listeners can easily learn and remember, other systems (for example earcons, see Section 10.6.1.1) use abstract sounds where the meanings are harder to learn. Problems did occur with some of the warning sounds used as Gaver et al. indicate: "the breaking bottle sound was so compelling semantically and acoustically that partners sometimes rushed to stop the sound without understanding its underlying cause or at the expense of ignoring more serious problems." Another problem was that when a machine ran out of raw materials its sound just stopped, users sometimes missed this and did not notice that something had gone wrong.

10.6.1.1 Design Guidelines for Auditory Icons

There have been few detailed studies investigating the best ways to design auditory icons, so there is little guidance for interaction designers. Mynatt (1994) has proposed the following basic design methodology: (1) choose short sounds that have a wide bandwidth, and where length, intensity, and sound quality are roughly equal. (2) Evaluate the identifiability of the auditory cues using free-form answers. (3) Evaluate the learnability of the auditory cues, which are not readily identified. (4) Test possible conceptual mappings for the auditory cues using a repeated measures design where the independent variable is the concept that the cue will represent. (5) Evaluate possible sets of auditory icons for potential problems with masking, discriminability, and conflicting mappings. (6) Conduct usability experiments with interfaces using the auditory icons.

10.6.2 Earcons

Blattner, Sumikawa, and Greenberg (1989) developed earcons. They use abstract, synthetic tones in structured combinations to create auditory messages. Blattner, Sumikawa, and Greenberg (1989) define earcons as "non-verbal audio messages that are used in the computer/user interface to provide information to the user about some computer object, operation, or interaction." Unlike auditory icons, there is no intuitive link between the earcon and what it represents; the link must be learned. They use a more traditional musical approach than auditory icons.

Earcons are constructed from simple building blocks called *motifs* (Blattner, Sumikawa, and Greenberg 1989). These are short, rhythmic sequences that can be combined in different ways. Blattner suggested their most important features:

Rhythm: Changing the rhythm of a motif can make it sound very different. Blattner, Sumikawa, and Greenberg (1989) describe this as the most prominent characteristic of a motif.

Pitch: There are 96 different pitches in the Western musical system, and these can be combined to produce a large number of different motifs.

Timbre: Motifs can be made to sound different by the use of different timbres, for example, playing one motif with the sound of a violin and the other with the sound of a piano.

Register: This is the position of the motif in the musical scale. A high register means a high-pitched note and a low register a low note. The same motif in a different register can convey a different meaning.

Dynamics: This is the volume of the motif. It can be made to increase as the motif plays (crescendo) or decrease (decrescendo).

Earcons can be constructed in two basic ways. The first, and simplest are *compound earcons*. These are simple motifs that can be concatenated to create more complex earcons. For example, a set of simple, one-element motifs might represent various system elements such as "create," "destroy," "file," and "string" (see Figure 10.5a), and these could then be concatenated to form earcons (Blattner, Sumikawa, and Greenberg 1989). In the figure, the earcon for "create" is a high-pitched sound that gets louder, for "destroy" it is a low-pitched sound that gets quieter. For "file," there are two long notes that fall in pitch and for "string" two short notes that rise. In Figure 10.5b, the compound earcons can be seen. For the "create file" earcon, the "create" motif is simply followed by the "file" motif. This provides a simple and effective method for building up earcons.

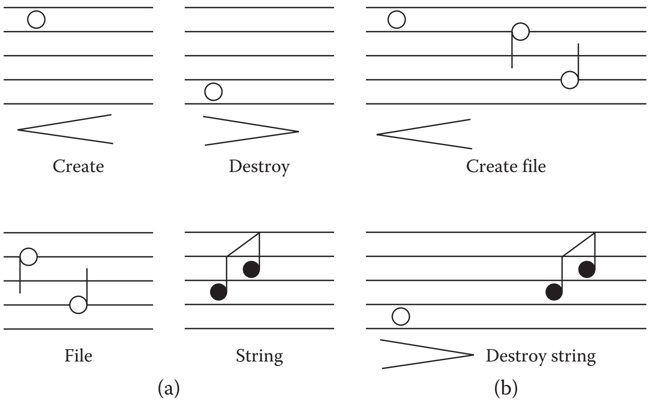


FIGURE 10.5 Compound earcons. (a) The four audio motifs "create," "destroy," "file," and "string" are shown. (b) The compound earcons "create file" and "destroy string" are shown. (From Blattner, M., D. Sumikawa, and R. Greenberg. 1989. *Hum Comput Interact* 4(1):11–44. With permission.)

Hierarchical earcons are more complex but can be used to represent more complex structures in sound. Each earcon is a node in a tree and inherits properties from the earcons above it. Figure 10.6 shows a hierarchy of earcons representing a family of errors. The top level of the tree is the family rhythm. This sound just has a rhythm and no pitch; the sounds used are clicks. The rhythmic structure of level 1 is inherited by level 2, but this time a second motif is added where pitches are put to the rhythm. At this level, Blattner et al. suggested that the timbre should be a sine wave, which produces a "colorless" sound. This is done so that at level 3 the timbre can be varied. At level 3, the pitch is also raised by a semitone to make it easier to differentiate from the pitches inherited from level 2. Other levels can be created where register and dynamics are varied.

Blattner et al. proposed the design of earcons but did not develop or test them. Brewster, Wright, and Edwards (1994) carried out a detailed evaluation of compound and hierarchical earcons based on the design proposed by Blattner, simple system beeps and a richer design based on more complex musical timbres using psychoacoustic research (see Section 10.3). In these experiments, participants were presented with earcons representing families of icons, menus, and combination of both (examples can be heard at www.dcs.gla.ac.uk/~stephen/demos.shtml). They heard each sound three times and then had to identify them when played back. Results showed that the more complex musical earcons were significantly more effective than both the simple beeps and Blattner's proposed design, with over 80% recalled correctly. Brewster et al. found that timbre was a much more important than that previously suggested, whereas pitch on its own was difficult to differentiate. The main design features of the earcons used were formalized into a set of design guidelines:

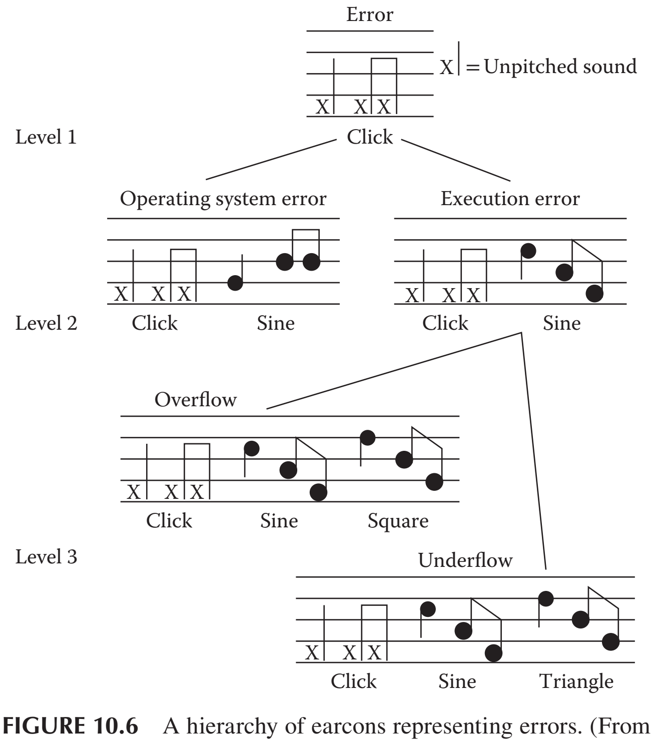


FIGURE 10.6 A hierarchy of earcons representing errors. (From Blattner, M., D. Sumikawa, and R. Greenberg. 1989. *Hum Comput Interact* 4(1):11–44. With permission.)

Timbre: This is the most important grouping factor for earcons. Use musical instrument timbres with multiple harmonics as this helps perception and can avoid masking. These timbres are more recognizable and differentiable.

Pitch and register: If listeners are to make absolute judgments of earcons, then pitch or register should not be used as a cue on its own. A combination of register and another parameter gives better rates of recall. If register alone must be used, then there should be large differences (two or three octaves) between earcons. Much smaller differences can be used if relative judgments are to be made. The maximum pitch used should be no higher than 5 kHz and no lower than 125–150 Hz so that the sounds are not easily masked and are within the hearing range of most listeners.

Rhythm, duration, and tempo: These make rhythms as different as possible. Putting different numbers of notes in each earcon is very effective. Earcons are likely to be confused if the rhythms are similar even if there are large spectral differences. Very short note lengths might not be noticed so do not use very short sounds. Earcons should be kept as short as possible so that they can keep up with interactions in the interface being sonified. Two earcons can be played in parallel to speed up presentation.

Intensity: This should not be used as a cue on its own because it is a major cause of annoyance. Earcons should be kept within a narrow dynamic range so that annoyance can be avoided (see Section 10.3 for more on this issue).

Major/minor mode: Lemmens (2005) showed that by changing from a major to minor key, he could change the affective responses of users to earcons. In Western music, the minor mode is broadly thought of as sad with the major as happy, and this can be used as a further cue to create differentiable earcons.

One aspect that Brewster also investigated was musical ability—as earcons are based on musical structures, is it only musicians who can use them? The results showed that the more complex earcons were recalled equally well by non-musicians as they were by musicians, indicating that they are useful to a more general audience of users.

In a further series of experiments, Brewster (1998b) looked in detail at designing hierarchical earcons to represent larger structures (with more than 30 earcons at four levels). These were designed based on the guidelines described earlier. Users were given a short training period and then were presented with sounds and they had to indicate where the sound was in the hierarchy.

Results were again good with participants recalling more than 80% correctly, even with the larger hierarchy used. The study also looked at the learning and memorability of earcons over time. Results showed that even with small amounts of training users could get good recall rates, and that the recall rates of the same earcons tested a week later was unchanged.

In a recent work, McGookin and Brewster (2004) have looked at presenting multiple earcons in parallel. This is problematic unless done carefully as the structures used to create earcons also cause them to overlap when played in parallel. McGookin suggested each earcon should have an onset delay and different spatial location to improve understanding. For examples of earcons in use, see the sonically enhanced widgets and Palm III work in Sections 10.7.1 through 10.7.3.

10.6.3 Sonification

Building on the work on accessibility comes from the idea of making data accessible. Sonification, or visualization in sound, can be used to present complex data nonvisually. There are many situations in which sonification can also be useful for sighted users (if they only have access to a small screen for example), or in combination with graphical feedback in multimodal visualization systems. Sonification is defined as "… the transformation of data relations into perceived relations in an acoustic signal for the purposes of facilitating communication or interpretation" (Kramer and Walker 1999). The range of sonification goes from the clicks of the Geiger counter to multidimensional information presentation of stock market data.

Mansur, Blattner, and Joy (1985) performed one of the most significant studies presenting data in sound. Their study, which laid out the research agenda for subsequent research in "sound graphs," used sound patterns to represent 2D line graphs. The value on the *y*-axis of the graph was mapped to pitch and the *x*-axis to time; this meant that a listener could hear the graph rise and fall over time in a similar way that a sighted person could see the line rising and falling. This is the basic technique used in most sonification systems.

They found that their approach was successful in allowing distinctions to be made between straight and exponential graphs, varying monotonicity in graphs, convergence, and symmetry. However, they did find that there were difficulties in identifying secondary aspects of graphs such as the slope of the curves. They suggested that a full sound graph system should contain information for secondary aspects of the graph such as the first derivative. Their suggestion was to encode this information by adding more overtones to the sound to change the timbre. They also suggested using special signal tones to indicate a graph's maxima or minima, inflection points, or discontinuities. Many other studies have been undertaken to develop this presentation technique further, most notably the studies by Flowers and Hauer (1992, 1995) and Walker (2002).

Walker and Cothran (2003) have produced the *Sonification Sandbox* to allow auditory interface designers to design sound graphs easily and rapidly. The software allows "… users to independently map several data sets to timbre, pitch, volume, and pan, with full control over the default, minimum, maximum, and polarity for each attribute." This gives the designers the chance to prototype sonifications, without having to create their own, custom-made applications. The software is freely available from http://sonify.psych.gatech.edu/research.

10.6.4 Tactons

Tactons (Brown and Brewster 2006) are structured vibrotactile messages that can be used to communicate information nonvisually. They are the tactile equivalent of earcons and visual icons, and could be used for communication in situations where vision is overloaded, restricted, or unavailable. Tactons are created by manipulating the parameters or dimensions of cutaneous perception to encode information. The most important dimensions (or parameters) are detailed below:

Locus: The body is a large area on which tactile actuators can be placed making locus (or spatial location) an important consideration. In his laboratory study, Geldard (1960) found that participants could reach levels of 100% recognition using seven actuators placed on the rib cage and the same results for five actuators on the chest. One issue that should be taken into account is the fact that, with standard vibrotactile actuators, the vibration emanates across the body and is not simply confined to underneath the actuator. Furthermore, when two or more actuators are activated simultaneously, it can often feel as though there is only one actuator.

Intensity: As mentioned in earcons, amplitude/intensity is not used as a parameter because users find loud sounds annoying and report annoyance when the volume level is out of their control. Using intensity as a parameter in tactons is equally problematic as reducing the amplitude could degrade perception of other parameters, or render the signal undetectable, while increasing it too far could cause pain (Geldard 1957). Therefore, it is best to leave amplitude under the control of the user instead of using it to encode information.

Duration: The duration parameter explored by Geldard (1957) is an extremely effective dimension. In his study, durations ranging between 0.1 and 2 seconds were used. 100% identification rates were achieved when four or five levels with intervals of at least 0.15 seconds were used. Differences in duration enable rhythmic structures to be created. However, it must be noted that stimulating an area of skin for an extended period can result in adaptation or even pain.

Frequency: As for tactile frequency, unfortunately humans cannot literally "hear through the skin" as the detectable frequency ranges for each modality are different (although with some overlap). Using frequency as a parameter has been difficult in experiments with issues rising from its influence on intensity perception. Frequency has yet to be used as a parameter in tacton research, but MacLean and Enriquez (2003) used multidimensional scaling techniques to determine how haptic icons can be created from signal parameters such as waveform, frequency, and force. They found that for the ranges of parameters that they implemented in a handheld knob, frequency played a dominant role in distinguishing between the multidimensional stimuli and that waveform and force were less helpful.

Waveform: Geldard (1957) suggested that it may be possible to distinguish between tactile waveforms provided the frequency of the stimuli is low and does not interfere. Musical composition studies have suggested that waveform can be correlated with the "texture" of tactile stimuli (Gunther, Davenport, and O'Modhrain 2002).

Rhythm: Rhythm is an extremely important parameter in earcon design and is the primary parameter used in tactons with recognition rates of over 90% achieved when three different rhythms are used (Brown, Brewster, and Purchase 2005). Rhythms can be created by grouping pulses of different durations. The rhythms used in tactons are based on Brewster's guidelines for rhythms in earcons (Brewster, Wright, and Edwards 1995).

10.6.4.1 Other Haptic Icons

Another approach to developing tactile or haptic icons involves identifying the basic elements, called haptic phonemes, and using these to create different haptic icons. With this method, Enriquez, MacLean, and Chita (2006) created a set of nine haptic icons that varied in terms of waveform and frequency. They then trained participants to associate each haptic icon with an arbitrary concept, such as the name of a fruit. They found that participants learned these associations after about 25 minutes of training and achieved higher identification rates with stimuli that varied in frequency (81% correct), compared with those that varied in waveform (73% correct).

Rovers and van Essen (2005) also mentioned the use of icons with haptic feedback. They stated that the message can be designed as a real-world signal such as a heartbeat or can be based on an abstract design. An abstract design requires the use of a set of common rules, for example, three pulses are equal to "off." In this case, variability can be represented in glyphs, for example, changing intensity based on running speed: the faster the speed, the higher the intensity.

10.6.5 Crossmodal Icons

Crossmodal interaction is a subset of multimodal interaction where the different senses are used to receive the same data. This provides a common representation of the data from both senses (in this case, audio and tactile) making them congruent informationally (Hoggan 2010b). Crossmodal use of the different senses allows the characteristics of one sensory modality to be transformed into stimuli for another sensory modality. Multimodal interaction, on the other hand, may also use the different senses to receive different information.

So, for example, in crossmodal interaction with a mobile device both the audio and tactile feedback would represent the same data, for example, an alarm. Whereas in multimodal interaction, the vibrotactile cue may indicate an alarm, the audio cue may represent a completely different type of information such as incoming messages.

The most popular example of crossmodal interaction can be seen in sensory substitution research. Sensory substitution systems take environmental data, which would normally be processed by one sensory system, and translate this data into stimuli for another sensory system (Lenay, Canu, and Villon 1997). The main application of these systems is increasing accessibility for those with sensory impairments. This class of systems includes tactile vision substitution, tactile auditory substitution, and teletouch.

Nonspeech audio and tactile displays are ideal candidates for crossmodal use because our senses of hearing and touch share several important similarities, in particular their temporal characteristics and their ability to perceive vibrations. An attribute that can communicate comparable information across modalities is considered to be amodal (Mendelson 1979). The shared temporal and spatial properties between audio and tactile mean that certain audio characteristics may be transformed into tactile stimuli (and *vice versa*) very easily (see Table 10.1). Therefore, the same data may be presented interchangeably via the two different modalities in crossmodal interaction depending on a user's particular disabilities or current situation.

Two icons may be considered to be crossmodal icons if and only if they provide a common representation of data, which is accessible interchangeably via different modalities (Hoggan 2010a).

As is the case with all types of icon including the earcons and tactons, for crossmodal icons to convey data successfully,

TABLE 10.1
Parameters Available in the Audio and Tactile Modalities

Parameter	Available in Audio?	Available in Tactile?
Rhythm	Yes	Yes
Pitch	Yes	No
Loudness (amplitude)	Yes	Yes
Timbre (texture)	Yes	Yes
Duration	Yes	Yes
Intensity	Yes	Yes
Spatial location	Yes	Yes
Rate (tempo)	Yes	Yes
Dynamics	Yes	No

there should be a mapping between the data to be communicated and the stimuli presented to the user. Crossmodal icons are structured abstract and use a symbolic approach as opposed to an iconic or indexical approach like those found in visual icons, auditory icons (William Gaver 1987), and hapticons (Mackenzie, Zhang, and Soukoreff 1999). These are not based on any preexisting understanding of the mapping between data and sound or touch. In other words, these mappings are arbitrary and require users to be trained to understand the relationship between data and sound or touch explicitly.

Crossmodal icons allow the same data to be accessible interchangeably via several different modalities. For example, a set of earcons/tactons can be considered to be crossmodal if the information represented can be encoded in both modalities so that users can move from an audio to a tactile presentation of the same data (and *vice versa*).

Multiple dimensions of data can be encoded in crossmodal icons, with each represented by a different crossmodal parameter. For example, if audio/tactile crossmodal icons were used to represent files in a computer interface, the file type could be represented by rhythm (in audio and tactile), size by duration (audio and tactile), and creation date by intensity (audio and tactile). Each file type would be mapped to a unique rhythm equivalent in both modalities. Therefore, two files of the same type and same size but different creation date would share the same audio/tactile rhythm and audio/tactile duration but would use different levels of audio/tactile intensity.

10.6.6 Comparing Feedback Types

Earcons and auditory icons are both effective at communicating information in sound. There is more formal evidence of this for earcons as more basic research has looked at their design. There is less basic research into the design of auditory icons, but the systems that have used them in practice have been effective. More detailed research is needed into auditory icons to correct this problem and to provide designers with design guidance on how to create effective sounds. It may be that each has advantages over the other in certain circumstances and that a combination of both is the best. In some situations, the intuitive nature of auditory icons may make them favorable. In other situations, earcons might be best because of the powerful structure they contain, especially if there is no real-world equivalent of what the sounds are representing. Indeed, there may be some middle ground where the natural sounds of auditory icons can be manipulated to give the structure of earcons.

The advantage of auditory icons over earcons is that they are easy to learn and remember because they are based on natural sounds and the sounds contain a semantic link to the objects they represent. This may make their association to certain, more abstract, actions or objects within an interface more difficult. Problems of ambiguity can also occur when natural sounds are taken out of the natural environment and context is lost (and people may also have their own idiosyncratic mappings). If the meanings of auditory icons must be learned then they lose some of their advantages and they come closer to earcons.

Earcons are abstract so their meaning must always be learned. This may be a problem, for example, in “walk up and use” type applications. Research has shown that little training is needed if the sounds are well designed and structured. Leplâtre and Brewster (2000) have begun to show that it is possible to learn the meanings implicitly while using an interface that generates the sounds as it is being used. However, some form of learning must take place. According to Blattner, Sumikawa, and Greenberg (1989), earcons may have an advantage when there are many highly structured sounds in an interface. With auditory icons, each one must be remembered as a distinct entity because there is no structure linking them together. With earcons, there can be a strong structure linking them that can easily be manipulated. There is not yet any experimental evidence to support this.

“Pure” auditory icons and earcons make up the two ends of a presentation continuum from representational to abstract. In reality, things are less clear. Objects or actions within an interface that do not have an auditory equivalent must have an abstract auditory icon made for them. The auditory icon then moves more toward the abstract end of the continuum. When hearing an earcon, the listener may hear and recognize a piano timbre, rhythm, and pitch structure as a kind of “catch-phrase” representing an object in the interface; he or she does not hear all the separate parts of the earcon and work out the meaning from them (listeners may also try and put their own representational meanings on earcons, even if the designer did not intend it as found by Brewster [1998b]). The earcon then moves more toward the representational side of the continuum. Therefore, earcons and icons are not necessarily as far apart as they might appear.

There are not yet any systems that use both types of sounds to their full extent and this would be an interesting area to investigate. Some parts of a system may have natural analogs in sound, and, therefore, auditory icons could be used; other parts might be more abstract or structured and earcons would be better (Figure 10.7). The combination of the two would be the most beneficial. This is an area ripe for further research.

The other major decision when choosing the type of nonvisual feedback to use is the choice of modality. If choosing audio, then Section 10.6.1 can inform the choice of earcons or auditory icons, both of which have been shown to be effective in communicating information through sound.

However, as mentioned in Section 10.2, audio feedback can be annoying if the volume is too loud or can go completely unnoticed if too quiet. Furthermore, there are social acceptability issues with audio feedback. It can be seen as rude to wear headphones when in the company of others

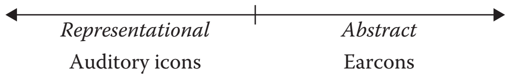


FIGURE 10.7 The presentation continuum of auditory icons and earcons.

(Hoggan and Brewster 2010). In these cases, it may be more advantageous to present feedback using the tactile modality due to its private and subtle nature.

When audio or tactile feedback is used in mobile situations, there are additional issues to consider. When the device is in a bag or pocket, tactile feedback can go unnoticed. When a user is in a noisy environment like public transport or listening to music, audio feedback can be ineffective. In these situations, there is a need for mobile devices to provide alternative presentation modalities through which information may be presented if the context requires. As the context changes, so should the feedback modality (guidelines can be found in the study by Hoggan et al. [2009]). By using concepts from crossmodal interaction and sensory substitution, mobile devices could translate data into an auditory or tactile form so that it can be presented in the most appropriate modality to suit the context. For example, alerts providing information to the user about incoming messages (for example, SMS, MMS, or phone call) could be encoded using crossmodal methods in both the audio and tactile modalities. By making these alerts available to both the auditory and tactile senses, users can receive the information in the most suitable way, without having to abandon their primary task to look at the device.

10.7 APPLICATIONS OF NONSPEECH FEEDBACK

Auditory and tactile output has been used in a wide range of different situations and applications. This section outlines some of the main areas of use and highlights some of the key papers in each area (for more uses of sound and touch, see the ICAD [www.icad.org], World Haptics [www.worldhaptics.org], or the Association for Computing Machinery international conference on Human factors in computing systems (ACM CHI) [www.acm.org/sigchi] series of conferences).

10.7.1 Graphical User Interfaces

One long-running strand of research in the area of auditory and tactile output is in the addition of sound and vibrations to standard graphical displays to improve usability. One reason for doing this is that users can become overloaded with visual information on large, high-resolution displays. In highly complex graphical displays, users must concentrate on one part of the display to perceive the visual feedback, so that feedback from another part may be missed. This becomes very important in situations where users must notice and deal with large amounts of dynamic data. For example, imagine you are working on your computer writing a report and are monitoring several on-going tasks such as a compilation, a print job, and downloading files over the Internet. The word-processing task will take up your visual attention because you must concentrate on what you are writing. To check when your printout is done, the compilation has finished, or the files have downloaded, you must move your visual attention away from the report and look at these other tasks. This causes the interface to intrude into the task you are trying to perform. If information about these other tasks were presented in sound or touch, then you could continue looking at the report but hear or feel information in the background about the other tasks.

One of the earliest pieces of work on sonic enhancement of an interface was Gaver's SonicFinder (1989). This used auditory icons to present information about the Macintosh interface redundantly with the graphical display.

Brewster (1998a) investigated the addition of sound to enhance graphical buttons. An analysis of the way buttons are used was undertaken highlighting some usability problems. It was found that the existing visual feedback did not indicate when mis-presses of a button might have occurred. For example, the selection of a graphical button is shown in Figure 10.8 (starting with 1.A and 2.A). The button highlights when it is pressed down (1.B and 2.B in Figure 10.8). There is no difference in feedback between a correct selection (1.C in Figure 10.8) and a mis-selection (2.C in Figure 10.8), where the user moves the mouse off the graphical button before the selection is complete. The user could therefore "slip off" the button, fail to press it, and get no feedback. This error can happen when the user is moving away from the button and on to some other task. For example, the user moves to a toolbar to press the "Bold" button and then moves back to the text to position the cursor to start typing. The button press and the mouse move overlap so that the button is not pressed. It is hard for the user to notice this because no feedback is given.

The problems could not easily be solved by adding more graphical feedback: The user is no longer looking at the button's location, so any feedback given there will be missed. Feedback could be given at the mouse location, but we cannot be sure that the user will be looking there either. Brewster designed a new button that used auditory feedback to indicate more about the state of the button. This was advantageous as sound is omnidirectional and the user does not need to focus attention on any part of the screen to perceive it.

Three earcons were used to improve the effectiveness of graphical buttons. An organ timbre was used for all of the sounds. When the user moved over a button, a continuous tone was played at 130 Hz at a volume just above the

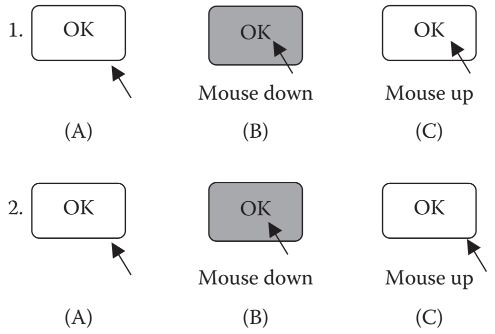


FIGURE 10.8 The visual feedback presented by a graphical button when selected. 1 shows a correct selection and 2 shows a slip-off. (From Brewster, S. A. 1998a. *Interact Comput* 11(2):211–35. With permission.)

background sound level. This informed the user the cursor was over the target (but could easily be habituated). When the mouse was pressed down over the graphical button, a continuous tone was played at 261 Hz. The third sound indicated that the graphical button had been successfully selected. This sound consisted of two short tones with a pitch of 1046 Hz and duration of 40 milliseconds. This sound was not played if a slip-off error occurred. If the user pressed the button very quickly, then only the success sound was played to avoid unnecessary feedback.

An experimental evaluation of these sounds was undertaken. Results showed that users recovered from slip-off errors significantly faster and with significantly fewer mouse clicks when sounds were present in the buttons. The users also significantly preferred the buttons with sound when asked to rate subjective preference. An interesting point to note was the fact that the use of no sound when a sound was expected could be attention grabbing. The participants could easily recognize a slip-off due to the demanding nature of the success sound *not* being played. This is important as reducing the amount of feedback presented is one way to make sure that it is not annoying.

Many other widgets have been successfully sonified. Beaudouin-Lafon and Conversey (1996) showed that nonspeech sounds could improve usability of scrollbars; Maury, Athenes, and Chatty (1999) and Marila (2002) added sounds to improve menu selections in drop-down menus. Ronkainen and Pasanen (2005) have done several studies into the design of audio feedback for buttons. Brewster and colleagues have investigated a wide range of different widgets including scroll bars, menus, progress bars, tool palettes, and drag and drop (Brewster 1998a). These widgets have been included in a toolkit (Crease, Gray, and Brewster 2000) that designers can use to add sound easily to their interfaces.

In terms of tactile feedback for widgets, Lee et al. (2004) created a system for providing tactile feedback for stylus-based touch-screen displays called the Haptic Pen. The Haptic Pen provides personal tactile feedback for multiple simultaneous users and can operate on large touch screens as well as ordinary surfaces. A pressure-sensitive stylus was combined with a small solenoid to generate a range of different tactile sensations. The tactile simulations generated by the Haptic Pen (the feedback when pressing a button appeared to feel realistic). The tactile feedback was intended to simulate the sensation of pressing a physical button or dragging a physical object. There was no formal study of the Haptic Pen so there can be few conclusions drawn but the initial findings indicate that the use of tactile feedback is an effective approach to simulating the sensation of pressing a physical button. Furthermore, the tactile feedback does not necessarily have to originate from the screen itself but can be incorporated into a stylus.

10.7.2 Users with Visual Impairments

One of the most important uses for sound and touch is in interfaces for people with visual disabilities. One of the main deprivations caused by blindness is the problem of access to information. A person with visual impairments will typically use a screen reader and a voice synthesizer to use a computer. The screen reader extracts textual information from the computer's video memory and sends it to the speech synthesizer to speak it. This works well for text but not well for the graphical components of current user interfaces. It is still surprising to find that many commercial applications used by people with visual impairments make little use of nonspeech sound or tactile feedback, concentrating on synthetic speech output. This is limiting (as discussed above) as speech is slow, can overload short-term memory, and is not good for presenting certain types of information; for example, it is not possible to render many types of images via speech so these can become inaccessible to people with visual impairments. One reason for the lack of use of sound and touch has been how to use it effectively, as Edwards (1995) says: "Currently the greatest obstacle to the exploitation of the variety of communications channels now available is our lack of understanding of how to use them." The combination of speech and nonspeech feedback such as earcons or tactons can increase the amount of information presented to the user. As long as this is done in a way that does not overload the user, then it can improve access to information. Some of the main research into the use of nonspeech interfaces for people with visual impairments will now be described (for more on perceptual impairments in general, see Chapter 38).

Soundtrack was an early attempt to create a word processor designed to be used by persons with visual impairments and was developed by Edwards (1989). It used earcons and synthetic speech as output and was designed so that the objects a sighted user would see in an interface, for example menus and dialogues, were replaced by auditory equivalents that were analogies of their visual counterparts. Its interface was constructed from auditory objects with which the user could interact. They were defined by a location, a name, a sound, and an action. They were arranged into a grid of two layers (see Figure 10.9) analogous to menus.

Each auditory object made a sound when the cursor entered it, and these could be used to rapidly navigate around the screen. Soundtrack used sine waves for its audio feedback. Chords were built-up for each menu depending on the number of menu items. For the edit menu, a chord of four notes was played because there were four menu items within it: cut, copy, paste, and find.

The base sounds increased in pitch from left to right—as in the normal representation of a musical scale (for example, on a piano), and the top layer used higher pitches than the bottom. Using these two pieces of information, a user could quickly find his or her position on the screen. If any edge of

File manu	Edit menu	Sound menu	Format menu
Alert	Dialog	Document 1	Document 2

FIGURE 10.9 Soundtrack's main screen. (From Edwards, A. D. N. 1989. *Hum Comput Interact* 4(1):45–66. With permission.)

the screen was reached, a warning sound was played. If at any point the user got lost or needed more precise information, he or she could click on an object and it would speak its name.

The approach taken in Soundtrack was to take the visual interface to a word processor and translate it into an equivalent auditory form. The *Mercator* system (Mynatt and Edwards 1995; Mynatt and Weber 1994) took a broader approach. The designers' goal was to model and translate the graphical interfaces of X Windows applications into sound without modifying the applications (and thus create a more general solution than Soundtrack's). Their main motivation was to simulate many of the features of graphical interfaces to make graphical applications accessible to users with visual impairments and keep coherence between the audio and visual interfaces so that users with visual impairments and sighted users could interact and work together on the same applications. This meant that the auditory version of the interface had to facilitate the same mental model as the visual one. This did not mean that they translated every pixel on the screen into an auditory form; instead, they modeled the interaction objects that were present. Modeling the pixels exactly in sound was ineffective due to the very different nature of visual and auditory media and the fact that graphical interfaces had been optimized to work with the visual sense (e.g., the authors claim that an audio equivalent of overlapping windows was not needed as overlapping was just an artifact of a small visual display). Nonspeech sound was an important aspect of their design to make the iconic parts of a graphical interface usable.

Mercator used three levels of nonspeech auditory cues to convey symbolic information presented as icons in the visual interface. The first level addressed the question of "what is this object?" In Mercator, the type of an interface object was conveyed with an auditory icon. For example, touching a window sounded like tapping a piece of glass, container objects sounded like a wooden box with a creaky hinge, and text fields used the sound of a manual typewriter. Although the mapping was easy for interface components such as trashcan icons, it was less-straightforward components that did not have simple referents in reality (e.g., menus or dialogue boxes, as discussed in Section 10.7.2). In Mercator, auditory icons were also parameterized to convey more detailed information about specific attributes such as menu length. Global attributes were also mapped into changes in the auditory icons. For example, highlighting and graying-out are common to a wide range of different widgets. To represent these, Mynatt et al. used sound filters. A low-pass filter was used to make the sound of a "grayed-out" object duller and more muffled.

Sensory substitution using the sense of touch has been used by persons with visual impairments for many years in the form of Braille (Lenay, Canu, and Villon 1997). In addition, vibrotactile displays have been developed to aid persons with visual impairments in accessing visual information such as text and pictures. There have been two main approaches to encoding information in these displays. The first two devices use a pictorial representation of the data: An image is captured by a camera and is reproduced directly as a pattern by vibrating the corresponding pins or actuators in a vibrotactile array. The third system uses a more abstract, coded approach to encode the information, where there is no direct relationship between the vibrotactile stimulus and the data that it represents and, therefore, the mapping between the two has to be learned. The final system uses an encoding scheme which is somewhere between these two approaches; some pictorial elements are retained, but where this is not possible some more abstract coding occurs.

The Optacon was developed in the 1960s as a commercially available reading device for people with visual impairments and was manufactured until production ceased in 1996. A small camera was moved over any material (text or graphics) that the user wished to read. The image captured by the camera was then presented to the user's fingertips through a 6 × 24 array of metal pins. This system used a direct, pictorial representation of the data, with the pins vibrating (at 230 Hz) to create a tactile reproduction of the image captured by the camera (detailed in the study by Craig and Sherrick [1982]). The Optacon was found to be reasonably effective, with reading speeds of around 10–12 wpm (words per minute) after the initial 9-day training period, reaching 30–50 wpm after further training and experience. Although these reading speeds are significantly slower than Braille, where the average reading speed for an adult is 104 wpm, the Optacon was beneficial for people with visual impairments as it allowed them to access any text or graphics without having to wait for it to be converted into Braille or tactile diagrams. The only parameter used to encode information in the Optacon display is the spatial pattern created by the pins. The user can control the speed of presentation by varying the speed at which they move the camera, but no other parameters are varied to encode data.

The Tactile Vision Substitution System, like the Optacon, converted images captured by a camera into vibrotactile patterns presented to the user's skin, with the aim of substituting vision (see the study by Craig and Sherrick [1982]). The user sat in a chair, which had a 20 × 20 array of vibrotactile transducers built into the back of it, and objects captured by the camera were presented by stimulating the transducers to make a pattern, which represented that object. These transducers were either on or off and could, therefore, only represent light and dark (nothing in between). Users were able to distinguish horizontal and vertical lines, and pick which object was being presented from a choice of 25, but had difficulty with the internal details of objects, such as facial features.

10.7.3 Mobile and Ubiquitous Computing

One of the major growth areas in computing at the beginning of the twenty-first century has been in mobile and ubiquitous computing. People no longer just use computers while sitting at a desk. Mobile telephones, PDAs, and handheld computers are now widely used (see Chapter 13 for more on mobile and ubiquitous computing). One problem with these devices is that there is a very limited amount of screen space on which to display information: The screens are small as the devices must be able to fit into the hand or pocket to be easily carried. Small screens can easily become cluttered with information and widgets and this presents a difficult challenge for interface designers.

The graphical techniques for designing interfaces on desktop interfaces do not apply well to handheld devices. Screen resources are limited; memory and processing power are much reduced from desktop systems. However, in many cases interface designs and interaction techniques have been taken straight from standard desktop graphical interfaces (where screen space and other resources are not a problem) and applied directly to mobile devices. This has resulted in devices that are hard to use, with small text that is hard to read, cramped graphics, and little contextual information. Audio and tactile feedbacks are an important way of solving these problems.

Another reason for using sound or touch is that if users are performing tasks while walking or driving, they cannot devote all of their visual attention to the mobile device. Visual attention must remain with the main task for safety. It is therefore hard to design a visual interface that can work well under these circumstances.

Brewster developed the ideas of sonified buttons described in Section 10.7.1 and applied them to buttons on the 3Com Palm series of pen-based handheld computers (Brewster 2002). Many of the same feedback problems with buttons apply in handhelds as in desktops, but are worse as the screen is smaller (and may be hard to see when the device is moving or the sun is shining). In addition, there is the problem of the stylus (or finger) obscuring the target on the display, which makes it difficult for users to know when they are pressing in the correct place. Simple earcons were used to overcome the problems. One aim of the work was to see if adding audio could reduce the size of the widgets so that screen space could be saved and to see the effects when users were on the move.

The results in general confirmed those of the previous study. Subjective workload in the sonically enhanced buttons was reduced compared with their silent counterparts. The addition of sound allowed the participants to enter significantly more five-digit strings than in the corresponding silent treatment, with smaller sonic buttons as effective as larger silent ones. When walking, there was a 20% drop in performance overall, with the sonic interface still performing better than the standard one. Participants walked further when sound was added, and the small buttons with sound allowed as much text to be entered as the large, silent buttons. The suggested reason for this was that users did not have to concentrate so much of their visual attention on the device, as much of the feedback needed was in sound, and so could look where they were going. This would therefore allow the size of items on the display to be reduced without a corresponding decrease in usability.

Sawhney and Schmandt (1999, 2000) developed a wearable computer-based personal messaging audio system called *Nomadic Radio* to deliver information and messages to users on the move. One of the aims of this system was to reduce the interruptions to a user caused by messages being delivered at the wrong time (e.g., mobile telephone calls being received in a meeting, a PDA beeping to indicate an appointment in the middle of a conversation). In the system, users wore a microphone and shoulder-mounted loudspeakers that provide a basic planar 3D audio environment (see Section 10.4.3) through which the audio was presented. A clock-face metaphor was used with 12:00 in front of the user's nose, 3:00 by the right ear, 6:00 directly behind the head, and so on. Messages were then presented in the position appropriate to the time that they arrived. The advantage of the 3D audio presentation (as described above) is that it allows users to listen to multiple simultaneous sound streams at the same time and still be able to distinguish and separate each one (the "cocktail party" effect [Arons 1992] shows that listeners can attend one stream of sound among many, but also monitor the others in case they need attention).

The system used a context-based notification strategy that dynamically selected the appropriate notification method based on the user's focus of attention. Seven levels of auditory presentation were used from silent to full speech rendering. If the user was engaged in a task, then the system was silent and no notification of an incoming call or message would be given (so as not to cause an interruption). The next level used "ambient" cues (based on auditory icons) with sounds like running water indicating that the system was operational. These cues were designed to be easily habituated but to let the user know that the system was working. The next level was a more detailed form of auditory cue giving information on system events, task completions, and mode transitions. For example, a ringing telephone sound was used to indicate the arrival of voicemail. These were more attention grabbing than the ambient cues and would only be played if the user was not fully occupied. The next four levels of cue used speech, expanding from a simple message summary up to the full text of a voicemail message. These might be used if the person wearing Nomadic Radio was not involved in tasks that required detailed attention. The system attempted to work out the appropriate level to deliver the notifications by listening to the background audio level in the vicinity of the user (using the built-in microphone) and if the user was speaking or not. For example, if the user was speaking, the system might use an ambient cue so as not to interrupt the conversation. Users could also press a button on the device to indicate they were busy and so turn it to silent.

Three-dimensional sound has been combined with gestures to create interactions where users can point at sound sources to choose them. An early example was from Cohen (1993) who created audio windows that users could manipulate with gestures, much as windows on a desktop computer could be controlled. Brewster et al. (2003) made this idea mobile and created a soundscape of audio sources around a listener's head that presented different types of information.

Users nodded at a sound source of interest to select it. A simple study showed that users could walk and nod to select items, but that there were many issues with sound placement and feedback. Further study by Marentakis looked at different types of gestures and feedback to improve the quality of mobile 3D audio interactions (Marentakis and Brewster 2004).

There has been much work in the area of notification systems using audio for ambient displays. Carefully designed nonspeech audio can grab attention and then fade into the background. An early example was *Audio Aura* by Mynatt et al. (1998), which aimed "to provide serendipitous information, via background auditory cues, that is tied to people's physical actions in the workplace." In a similar way to Nomadic Radio, Audio Aura used auditory icons to provide background information that did not distract users.

The system used active badges so that the location of users could be identified and appropriate audio cues given, along with wireless headphones so that users could hear the sounds without distracting others. The location information from the active badges was combined with other data sources such as on-line calendars and e-mails. Changes in this information triggered audio cues sent to the user through the headphones.

Here are some examples of how the system might be used. In the first example, the user goes to the office coffee room and while entering the room he or she hears information about the number and type of e-mail messages currently waiting. This would give the user a cue whether to stay and talk to their colleagues or go back to the office to answer the messages. In the second example, a user goes to his or her colleague's office but the occupant is not there. Audio Aura would play sounds indicating if the occupant has been in recently or away for a longer period. The authors were keen to make sure the sounds were not distracting and attention grabbing—they were meant to give background information and not to be alarms. To this end, great care was taken with the cue design. They attempted to design "sonic ecologies"—groups of sounds that fitted together into a coherent whole. For example, one set of cues was based on a beach scene. The number of new e-mails was mapped to seagull cries: The more e-mails the more the gulls cried. Group activity levels were mapped to the sound of surf: The more activity going on within the group the more active the waves became. These cues were very subtle and did not grab users' attention, but some learning of the sounds would be needed as they are quite abstract.

This work was taken on and implemented in a realistic environment by Kilander and Lonnqvist (2001, 2002). They created a *Weakly Intrusive Ambient Sound scape* (WISP) where states in the computational or physical ubiquitous computing environment are presented as subtle, nonintrusive sound cues based on auditory icons, with each cue "sufficiently nonintrusive to be accepted without disturbing the focus of the task at hand, while distinctive enough to be separable from other cues." They described a meeting room scenario where devices such as handheld computers, public PCs, and clocks might all be able to make sounds and give cues for ambient awareness. The level of intrusiveness of the sounds could be varied. For low intrusiveness, a quiet sound was played with lots of reverb, making the cue sound far away and almost inaudible, whereas for high intrusiveness sharp sounds were played with no reverb.

One problem with the system was choice of sounds; users could be detected by the environment and their personal sound mappings and parameters chosen. However, these mappings could conflict with others' choices; for example, two users might use the same sound cue for different events. This could be solved to some extent with local presentation of the sounds, as in Nomadic Radio, but global cues would be more of a problem for these types of systems.

Earcons have also been used to provide previews of information. Shirazi et al. (2010) introduced the concept of audio previews of SMS. Based on a real-time analysis of the content of a message, auditory cues were provided in addition to the notification tone upon receiving an SMS. A field study showed that the use of audio-enhanced SMS affects the reading and writing behavior of users. There was a significant impact when it comes to checking messages in situations where users are engaged in other activities. For example, question marks often led users to check messages immediately.

Shengdong et al. (2007) created earPod, a touch-based menu technique with reactive auditory feedback through simple stereo audio based on time and intensity differences between the ears. The results of a user study showed that earPod is efficient to use and relatively easy to learn. For fairly large static menus, the earPod method was comparable in both speed and accuracy with an iPod-like visual menu selection technique. Furthermore, although initially slower, earPod outperformed the visual technique within 30 minutes of training.

Many mobile interfaces have also been successfully augmented with tactile feedback, which may be appropriate at times when audio feedback could go unnoticed such as in noisy environments. Nashel and Razzaque (2003) added tactile cues simulating real buttons to virtual buttons displayed on mobile devices with touch screens. Hoggan, Brewster, and Johnston (2008) also studied the use of tactile feedback for text entry in mobile environments. The experiment compared devices with a physical keyboard, a standard touch screen, and a touch screen with tactile feedback added in both static and mobile environments. The results showed that the addition of tactile feedback to the touch screen significantly improved finger-based text entry, bringing it close to the performance of a real physical keyboard. A second experiment showed that higher specification tactile actuators could improve performance even further with fewer errors and greater speeds of text entry compared with standard touchscreen keyboards without tactile feedback.

One advantage of using tactile feedback is that actuators can be placed all over the body of the user. Lee and Stamer (Lee and Thad 2010) presented Buzz wear, a tactile display for the wrist, which provides alerts for mobile users. The first experiment focused on the perception sensitivity of tactile patterns and showed that people can discriminate 24 tactile patterns on the wrist with up to 99% accuracy after 40 minutes of training. Among the four parameters (intensity, starting point, temporal pattern, and direction) that vary in the 24 patterns, intensity is the most difficult parameter to distinguish and temporal pattern is the easiest. The second experiment focused on dual-task performance, exploring users' abilities to perceive three incoming alerts from two mobile devices with and without visual distraction. The results showed that, when visually distracted, users' reactions to incoming alerts become slower for the standard mobile phone but not for the wrist-based tactile display.

10.7.4 Crossmodal Applications

Despite the fact that research has shown both audio and tactile icons to be effective means of communication, the area of crossmodal auditory/tactile displays has been studied less. Recently, Immersion Corporation has created VibeTonz (http://www.immersion.com), which could be considered crossmodal. These are vibrotactile messages that can be used, like personalized ringtones, to indicate the identity of the caller in a mobile phone. However, there have been no empirical tests conducted to determine the effectiveness of these cues or to discover the amount of information that could be encoded in the cues.

The Touch Engine by Sony is another system that could be considered crossmodal. Sony's Touch Engine has a vibrotactile screen through which users can feel images and buttons that are on the screen. In the Touch Engine, a heart icon is represented by a heartbeat sensation (Poupyrev, Rekimoto, and Maruyama 2002). Although this is not a direct translation from vision, it is an intuitive, direct translation from sound and, in fact, the sense of touch itself (as you can feel someone's heartbeat).

Chang and O'Sullivan (2005) are some of the small number of researchers who have used both the audio and tactile modalities in a mobile device. In the most basic terms, tactile feedback is added to enhance the audio feedback in a standard mobile device. The authors argue that by using integrated stimulation of the five basic senses, the sense of cognition is engaged more fully. The authors present techniques for audio manipulation to create simple vibrotactile feedback based on the fact that both the audio and tactile modalities are made up of vibrations. A filter is applied to split the sound into its constituents, that is, vibrotactile and audio. In this case, any frequencies below 300 Hz were amplified and presented through the tactile actuators. Frequencies above this level were presented through audio. Although in this case the tactile feedback is used purely as an enhancement to the audio modality, the crossmodal similarities between the modalities are exploited through the use of frequency.

Lastly, Hoggan and Brewster (2010) conducted a longitudinal summative evaluation of a touch-screen application, CrossTrainer, which uses crossmodal earcons and tactons on a mobile touch-screen device. The aim was to investigate the everyday use of crossmodal audio and tactile feedback and to study user performance and preference over time. The results of the study showed that crossmodal feedback aids users in entering answers quickly and accurately using a variety of different widgets. Furthermore, the results demonstrated that users can switch between modalities and reach 100% recognition rates of multidimensional crossmodal alerts after 2 days of regular use. Overall, the CrossTrainer study highlighted issues to consider when choosing between audio and tactile feedback for a mobile touch-screen application. The results indicated that environmental noise and vibration levels, personal preference, location, and period of use should be taken into account when choosing between nonspeech audio and tactile feedback.

10.8 CONCLUSIONS

Research into the use of nonspeech sounds and tactile feedback for information display has shown its benefits in a wide range of different applications from systems for blind people to ubiquitous computing. There are many good examples that designers can look at to see how sound or touch may be used effectively and design guidelines are now starting to appear.

Three areas are likely to be important in its future growth. The first is in combining the sound and tactile modalities with others (vision, force-feedback, etc.) to create multimodal or crossmodal displays that make the most of the all the senses available to users. This is an area ripe for further investigation, and there are many interesting interaction problems that can be tackled when multiple senses are used together. The second area in which crossmodal audio and tactile feedback could play a major role is with multitouch and tabletop interfaces. In these cases, feedback will be required for separate fingers and also separate users. Many large touch-screen computers use direct finger-based multitouch input and a 360-degree user experience. This configuration means that users should be able to use the table without restriction no matter where they are positioned around it. Both audio and tactile feedback could be advantageous over visual feedback alone. The third area in which nonspeech sound and tactile feedback has a large part to play is with mobile/wearable computing devices (again also in a multimodal or crossmodal form). Small screens cause many difficult presentation problems, and this is exactly the situation in which sound or touch has many advantages—they do not take up any precious screen space and users can hear or feel it even if they cannot look at their device. In a ubiquitous setting, there may not even be a screen at all and sound or tactile feedback can provide information on the services available in a particular environment in a nonintrusive way.

REFERENCES

Arons, B. 1992. A review of the cocktail party effect. *J Am Voice I/O Soc* 12:35–50.

Bagwell, C. 1998. Audio file formats FAQ. http://www.cnpbagwell.com/audio.html (accessed November 2005).

Beaudouin-Lafon, M., and S. Conversy. 1996. Auditory illusions for audio feedback. Paper presented at the ACM CHI '96 Conference Companion, Vancouver, Canada.

Berglund, B., K. Harder, and A. Preis. 1994. Annoyance perception of sound and information extraction. *J Acoust Soc Am* 95(3):1501–9.

Blattner, M., and R. B. Dannenberg, eds. 1992. *Multimedia Interface Design*. New York: ACM Press, Addison-Wesley.

Blattner, M., D. Sumikawa, and R. Greenberg. 1989. Earcons and icons: Their structure and common design principles. *Hum Comput Interact* 4(1):11–44.

Blauert, J. 1997. *Spatial Hearing*. Cambridge, MA: MIT Press.

Bly, S. 1982. *Sound and Computer Information Presentation* (Unpublished PhD Thesis No. UCRL53282: Lawrence Livermore National Laboratoryo. Document Number).

Bolanowski, S. J., G. A. Gescheider, R. T. Verrillo, and C. M. Chechosky. 1988. Four channels mediate the mechanical aspects of touch. *J Acoust Soc Am* 84(5):1680–94.

Brewster, S. A. 1998a. The design of sonically-enhanced widgets. *Interact Comput* 11(2):211–35.

Brewster, S. A. 1998b. Using non-speech sounds to provide navigation cues. *ACM Trans Comput Hum Interact* 5(3):224–59.

Brewster, S. A. 2002. Overcoming the lack of screen space on mobile computers. *Pers Ubiquitous Comput* 6(3):188–205.

Brewster, S. A., J. Lumsden, M. Bell, M. Hall, and S. Tasker. 2003. Multimodal 'Eyes-Free' interaction techniques for wearable devices. Paper presented at the Proceedings of ACM CHI 2003, Fort Lauderdale, FL.

Brewster, S. A., S. Wall, L. M. Brown, and E. Hoggan. 2008. Tactile displays. In *The Engineering Handbook on Smart Technology for Aging, Disability and Independence*, ed. A. Helal, M. Mokhtari, and B. Abdulrazak. New York: John Wiley & Sons.

Brewster, S. A., P. C. Wright, and A. D. N. Edwards. 1994. A detailed investigation into the effectiveness of earcons. In *Auditory Display*, ed. G. Kramer, 471–98. Reading, MA: Addison-Wesley.

Brewster, S. A., P. C. Wright, and A. D. N. Edwards. 1995. Experimentally derived guidelines for the creation of earcons. Paper presented at the HCI '95. Huddlesfield, UK: BCS.

Brown, L. M., and S. A. Brewster. 2006. Multidimensional tactons for non-visual information display in mobile devices. Paper presented at the MobileHCI '06. New York: ACM.

Brown, L. M., S. A. Brewster, and H. C. Purchase. 2005. A first investigation into the effectiveness of tactons. Paper presented at the WorldHaptics 2005, Pisa, 167–76Y. New York: IEEE.

Buxton, W. 1989. Introduction to this special issue on nonspeech audio. *Hum Comput Interact* 4(1):1–9.

Buxton, W., W. Gaver, and S. Bly. 1991. Tutorial number 8: The use of non-speech audio at the interface. Paper presented at the Proceedings of ACM CHI '91, New Orleans, LA.

Chang, A., S. O'Modhrain, R. Jacob, E. Gunther, and H. Ishii. 2002. ComTouch: Design of a vibrotactile communication device. Paper presented at the Designing Interactive Systems, London, UK.

Chang, A., and C. O'Sullivan. 2005. Audio-haptic feedback in mobile phones. Paper presented at the CHI '05 extended abstracts on Human factors in computing systems, Portland, OR.

Cholewiak, R. W., J. C. Brill, and A. Schwab. 2004. Vibrotactile localization on the abdomen: Effects of place and space. *Percept Psychophys* 66:970–87.

Cholewiak, R. W., and A. A. Collins. 1995. Vibrotactile pattern discrimination and communality at several body sites. *Percept Psychophys* 57(5):724–37.

Cholewiak, R. W., and J. C. Craig. 1984. Vibrotactile pattern recognition and discrimination at several body sites. *Percept Psychophys* 35:503–14.

Chowning, J. 1975. Synthesis of complex audio spectra by means of frequency modulation. *J Audio Eng Soc* 21(7):526–34.

Cohen, M. 1993. Throwing, pitching and catching sound: Audio windowing models and modes. *Int J Man Mach Stud* 39:269–304.

Craig, J. C., and C. E. Sherrick. 1982. Dynamic tactile displays. In *Tactual Perception: A Sourcebook*, ed. W. Schiff and E. Foulke, 209–33. Cambridge, UK: Cambridge University Press.

Crease, M. C., P. D. Gray, and S. A. Brewster. 2000. Caring, sharing widgets. Paper presented at the Proceedings of BCS HCI 2000, Sunderland, UK.

Edwards, A. D. N. 1989. Soundtrack: An auditory interface for blind users. *Hum Comput Interact* 4(1):45–66.

Edwards, A. D. N., ed. 1995. *Extra-Ordinary Human-Computer Interaction*. Cambridge, UK: Cambridge University Press.

Edworthy, J., S. Loxley, and I. Dennis. 1991. Improving auditory warning design: Relationships between warning sound parameters and perceived urgency. *Hum Factors* 33(2):205–31.

Enriquez, M., K. MacLean, and C. Chita. 2006. Haptic phenomes: Basic building blocks of haptic communication. Paper presented at the 8th International Conference on Multimodal Interfaces (ICMI '06). New York: ACM.

Flowers, J. H., and T. A. Hauer. 1992. The ear's versus the eye's potential to assess characteristics of numeric data: Are we too visuocentric? *Behav Res Methods Instrum Comput* 24:258–64.

Flowers, J. H., and T. A. Hauer. 1995. Musical versus visual graphs: Cross-modal equivalence in perception of time series data. *Hum Factors* 37:553–69.

Frysinger, S. P. 1990. Applied research in auditory data representation. Paper presented at the Extracting meaning from complex data: processing, display, interaction. Proceedings of the SPIE/SPSE symposium on electronic imaging, Springfield, MA.

Gaver, W. 1987. Auditory icons: Using sound in computer interfaces. *ACM SIGCHI Bull* 19(1):74.

Gaver, W. 1989. The SonicFinder: An interface that uses auditory icons. *Hum Comput Interact* 4(1):67–94.

Gaver, W. 1997. Auditory interfaces. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. Landauer, and P. Prabhu., 2nd ed., 1003–42. Amsterdam: Elsevier.

Gaver, W., R. Smith, and T. O'Shea. 1991. Effective sounds in complex systems: The ARKola simulation. Paper presented at the Proceedings of ACM CHI '91, New Orleans, LA.

Geldard, F. A. 1957. Adventures in tactile literacy. *Am Psychol* 12:115–24.

Geldard, F. A. 1960. Some neglected possibilities of communication. *Science* 131(3413):1583–8.

Gelfand, S. A. 1981. *Hearing: An Introduction to Psychological and Physiological Acoustics*. New York: Marcel Dekker Inc.

Goff, G. D. 1967. Differential discrimination of frequency of cutaneous mechanical vibration. *J Exp Psychol* 74:294–9.

Gunther, E., G. Davenport, and S. O'Modhrain. 2002. Cutaneous grooves: Composing for the sense of touch. Paper presented at the New Interfaces for Musical Expression, Dublin, Ireland.

Hoggan, E. 2010a. *Crossmodal Audio and Tactile Interaction with Mobile Touchscreens*. Glasgow: University of Glasgow.

Hoggan, E. 2010b. Crossmodal Audio and Tactile Interaction with Mobile Touchscreens: Thesis Summary. *IJMHCI*, 29–44, Pennsylvania (USA): IGI Global.

Hoggan, E., and S. A. Brewster. 2010. CrossTrainer: Testing the use of multimodal interfaces in situ. Paper presented at the CHI '10, Atlanta, Georgia.

Hoggan, E., S. A. Brewster, and J. Johnston. 2008. Investigating the effectiveness of tactile feedback for mobile touchscreens. Paper presented at the CHI '08, Florence, Italy.

Hoggan, E., A. Crossan, S. A. Brewster, and T. Kaaresoja. 2009. Audio or tactile feedback: Which modality when? Paper presented at the CHI '09, 2253–2256. Boston, MA, USA: ACM Press.

ISO: Ergonomics of human-computer interaction—Part 910: Framework for tactile and haptic interaction. 2009. Retrieved. from http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=51097

Kaczmarek, K. A., and P. Bach-Y-Rita. 1995. *Tactile Displays*. New York: Oxford University Press.

Kandel, E. R., and T. M. Jessell. 1991. Touch. In *Principles of Neural Science*, ed. E. R. Kandel, J. H. Schwartz, and T. M. Jessell, 349–414. New York: Oxford University Press.

Kilander, F., and P. Lonnqvist. 2001. A weakly intrusive ambient soundscape for intuitive state perception. In *Continuity in Future Computing Systems*, ed. G. Doherty, M. Massink, and M. Wilson, 70–4. Oxford: CLRC.

Kilander, F., and P. Lonnqvist. 2002. A whisper in the woods—an ambient soundscape for peripheral awareness of remote processes. Paper presented at the Proceedings of ICAD 2002, Kyoto, Japan.

Koskinen, E., T. Kaaresoja, and P. Laitinen. 2008. Feel-good touch: Finding the most pleasant tactile feedback for a mobile touch screen button. Paper presented at the ICMI 2008, 297–304. Crete: ACM Press.

Kramer, G. 1994a. An introduction to auditory display. In *Auditory Display*, ed. G. Kramer, 1–77. Reading, MA: Addison-Wesley.

Kramer, G., ed. 1994b. *Auditory Display* (Vol. Proceedings volume XVIII). Reading, MA: Addison-Wesley.

Kramer, G., and B. Walker, eds. 1999. *Sonification Report: Status of the Field and Research Agenda*. Santa Fe, NM: The International Community for Auditory Display.

Laitinen, P., and J. Mäenpää. 2006. Enabling mobile haptic design: Piezoelectric actuator technology properties in handheld devices. Paper presented at the HAVE '06, 40–43. Ottawa, Ont, Canada: IEEE

Lee, J. C., P. H. Dietz, D. Leigh, W. S. Yerazunis, and S. E. Hudson. 2004. Haptic pen: A tactile feedback stylus for touch screens. Paper presented at the 17th annual ACM symposium on User interface software and technology, 291–294. Santa Fe, New Mexico: ACM Press.

Lee, S., and S. Thad. 2010. BuzzWear: Alert perception in wearable tactile displays on the wrist. Paper presented at the Proceedings of the 28th international conference on Human factors in computing systems. 433–442. Atlanta, Georgia: ACM Press.

Lemmens, P. 2005. Using the major and minor mode to create affectively-charged earcons. Paper presented at the Proceedings of ICAD 2005, Limerick, Ireland.

Lenay, C., S. Canu, and P. Villon. 1997. Technology and perception: The contribution of sensory substitution systems. Paper presented at the ICCT. Aizu, Japan: IEEE

Leplâtre, G., and S. A. Brewster. 2000. Designing non-speech sounds to support navigation in mobile phone menus. Paper presented at the Proceedings of ICAD 2000, Atlanta, GA.

Mackenzie, I. S., S. Zhang, and R. W. Soukoreff. 1999. Text entry using soft keyboards. *Behav Inf Technol* 18(4):235–44.

MacLean, K., and M. Enriquez. 2003. Perceptual design of haptic icons. Paper presented at the Eurohaptics, Dublin, Ireland.

Mansur, D. L., M. Blattner, and K. Joy. 1985. Sound-Graphs: A numerical data analysis method for the blind. *J Med Syst* 9:163–74.

Marentakis, G., and S. A. Brewster. 2004. A study on gestural interaction with a 3D audio display. Paper presented at the Proceedings of MobileHCI 2004, Glasgow, UK.

Marila, J. 2002. Experimental comparison of complex and simple sounds in menu and hierarchy sonification. Paper presented at the Proceedings of ICAD 2003, Kyoto, Japan.

Maury, S., S. Athenes, and S. Chatty. 1999. Rhythmic menus: Toward interaction based on rhythm. Paper presented at the Extended Abstracts of ACM CHI'99, Pittsburgh, PA.

McCormick, E. J., and M. S. Sanders. 1982. *Human Factors in Engineering and Design.* 5th ed. New York: McGraw-Hill.

McGookin, D. K., and S. A. Brewster. 2004. Understanding concurrent earcons: Applying auditory scene analysis principles to concurrent earcon recognition. *ACM Trans Appl Percept* 1(2):120–55.

Mendelson, M. J. 1979. Acoustic-optical correspondences and auditory-visual coordination in infancy. *Can J Exp Psychol* 33:334–46.

Miranda, E. R. 1998. *Computer Sound Synthesis for the Electronic Musician*. Oxford, UK: Focal Press.

Moore, B. C. 2003. *An Introduction to the Psychology of Hearing.* 5th ed. Oxford, UK: Elsevier Science.

Mursell, J. L. 1937. *The Psychology of Music*. New York: W.W. Norton and Company, Inc.

Mynatt, E. D. 1994. Designing with auditory icons: How well do we identify auditory cues? Paper presented at the Proceedings of the CHI '94 conference companion, Boston, MA.

Mynatt, E. D., M. Back, R. Want, M. Baer, and J. B. Ellis. 1998. Designing audio aura. Paper presented at the Proceedings of ACM CHI '98, Los Angeles, CA.

Mynatt, E. D., and K. Edwards. 1995. Metaphors for non-visual computing. In *Extra-Ordinary Human-Computer Interaction*, ed. A. D. N. Edwards, 201–20. Cambridge, UK: Cambridge University Press.

Mynatt, E. D., and G. Weber. 1994. Nonvisual presentation of graphical user interfaces: Contrasting two approaches. Paper presented at the Proceedings of ACM CHI '94, Boston, MA.

Nashel, A., and S. Razzaque. 2003. Tactile virtual buttons for mobile devices. Paper presented at the CHI '03 extended abstracts, Ft. Lauderdale, FL.

Neuhoff, J. G., ed. 2004. *Ecological Psychoacoustics*. San Diego, CA: Elsevier Academic Press.

Patterson, R. D. 1989. Guidelines for the design of auditory warning sounds. *Proc Inst Acoust Spring Conf* 11(5):17–24.

Phillips, J. R., and K. O. Johnson. 1985. Neural mechanisms of scanned and stationary touch. *J Acoust Soc Am* 77:220–4.

Pohlmann, K. 2005. *Principles of Digital Audio.* 5th ed. New York: McGraw-Hill.

Poupyrev, I., J. Rekimoto, and S. Maruyama. 2002. TouchEngine: A tactile display for handheld devices. Paper presented at the ACM CHI 2002, MN.

Roads, C. 1996. *The Computer Music Tutorial*. Cambridge, MA: MIT Press.

Ronkainen, S., and L. Pasanen. 2005. Effect of Aesthetics on audio-enhanced graphical buttons. Paper presented at the Proceedings of ICAD 2005, Limerick, Ireland.

Rovers, A. F., and H. A. van Essen. 2005. Guidelines for haptic interpersonal communication applications: An exploration of foot interaction styles. *Virtual Real* 9(2):177–91.

Sawhney, N., and C. Schmandt. 1999. Nomadic radio: Scalable and contextual notification for wearable messaging. Paper presented at the Proceedings of ACM CHI '99, Pittsburgh, PA.

Sawhney, N., and C. Schmandt. 2000. Nomadic radio: Speech and audio interaction for contextual messaging in nomadic environments. *ACM Trans Hum Comput Interact* 7(3):353–83.

Shengdong, Z., D. Pierre, C. Mark, B. Ravin, and B. Patrick. 2007. Earpod: Eyes-free menu selection using touch input and reactive audio feedback. Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, 1395–1404. San Jose, CA: ACM Press.

Sherrick, C. E., and R. W. Cholewiak. 1986. Cutaneous sensitivity. In *Handbook of Perception and Human Performance*, ed. K. Boff, L. Kaufman, and J. L. Thomas, 1–58. New York: Wiley & Sons.

Sherrick, C. E., R. W. Cholewiak, and A. A. Collins. 1990. The localization of low- and high-frequency vibrotactile stimuli. *J Acoust Soc Am* 88(1):169–79.

Shirazi, A. S., A. H. Sarjanoja, F. Alt, A. Schmidt, and J. Hkkil. 2010. Understanding the impact of abstracted audio preview of SMS. Paper presented at the Proceedings of the 28th international conference on Human factors in computing systems, 1735–1738. Atlanta, GA: ACM Press.

Strybel, T., C. Manligas, and D. Perrott. 1992. Minimum audible movement angle as a function of the azimuth and elevation of the source. *Hum Factors* 34(3):267–75.

Summers, I. R. 1992. *Tactile Aids for the Hearing Impaired*. London, UK: Whurr Publishers Ltd.

Summers, I. R., P. R. Dixon, and P. G. Cooper. 1994. Vibrotactile and electrotactile perception of time-varying pulse trains. *J Acoust Soc Am* 95(3):1548–58.

Terhardt, E. 1974. On the perception of periodic sound fluctuations (Roughness). *Acustica* 30:201–13.

Thimbleby, H. 1990. *User Interface Design*. New York: ACM Press, Addison-Wesley.

van Veen, H. A. H. C., and J. B. van Erp. 2000. Tactile information presentation in the cockpit. Paper presented at the First International Workshop on Haptic Human-Computer Interaction.

Verrillo, R. T. 1966. Vibrotactile thresholds for hairy skin. *J Exp Psychol* 72(1):47–50.

Verrillo, R. T., and G. A. Gescheider. 1992. Perception via the sense of touch. In Tactile Aids for the Hearing Impaired, ed. I.R. Summers, 1–36. London: Whurr Publishers.

Walker, B. N. 2002. Magnitude estimation of conceptual data dimensions for use in sonification. *J Exp Psychol Appl* 8:211–21.

Walker, B. N., and J. Cothran. 2003. Sonification sandbox: A graphical toolkit for auditory graphs. Paper presented at the Proceedings of ICAD 2003, Boston, MA.

11 Network-Based Interaction

Alan Dix

CONTENTS

11.1 INTRODUCTION

In some ways, this chapter could be seen as redundant in a human–computer interaction (HCI) book—surely networks are just an implementation mechanism, a detail below the surface, and all that matters are the interfaces that are built on them. However, networked interfaces, especially the web, but increasingly also mobile devices, have changed the way we view the world and the way we view the society. Even those bastions of conservatism, the financial institutions have found themselves in sea-change and a complete restructuring of the fundamentals of businesses … just an implementation detail! Indeed networks have become so ubiquitous, so enmeshed in our day-to-day lives that it is becoming hard to distinguish to envisage interaction that is not network-based.

11.1.1 Structure

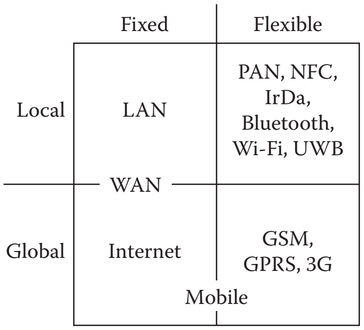
The chapter will begin with a brief overview of types of networks (Section 11.2), focused on the network as technical infrastructure. It then deals with network-based interaction under four main headings:

- Networks as enablers (Section 11.3): things that are only possible with networks
- Networks as mediators (Section 11.4): issues and problems because of networks
- Networks as subjects (Section 11.5): understanding and managing networks
- Networks as platforms (Section 11.6): algorithms and architectures for distributed interfaces

In addition, there will be a section (Section 11.7) that takes a broader view of the history and future of network interaction and the societal effects and paradigm changes engendered, especially by more recent developments in global and wireless networking.

11.2 ABOUT NETWORKS

The word network will probably make many think of accessing the Internet and the web. Others may think of a jumble of Ethernet wires between the PCs in their office, maybe broadband router at home, or Wi-Fi hotspots in cafes. In fact, the range of networking standards including physical cabling (or lack of cabling) and the protocols that computers use to talk down those cables is extensive. Although most of the wire-based networks have been around for some time, they are in a state of flux as a result of increases in scale and the demands of continuous media. In the wireless world, things are changing even more rapidly with new generations of data service being introduced every few years.



As an aid to seeing the broader issues surrounding these changing (and, in some cases potentially ephemeral) technologies, we can use the following two dimensions to classify them:

- Global versus local
  - How spatially distant are the points connected—ranging from machines in the same room (IrDa, Bluetooth), through those in a building/site (local area network [LAN]) to global networks (Internet, mobile-phone networks).

- Fixed versus flexible
    - How permanent are the links between points of the network, from physically fixed machines, to self-reconfiguring devices that recognize other devices in their vicinity.

The fixed versus flexible dimension is almost, but not quite terrestrial versus wireless. The “not quite” is because fixed networks increasingly involve wireless links. Also, it is often possible, when visiting another organization, to plug a portable computer into a (wired) Ethernet network and find you have access to the local printers, Internet connections, and so on—flexible wire-based networking.

Let us look at a few network technologies against these dimensions. Traditional office LANs are squarely in the local-fixed category, whereas the Internet is largely global-fixed category. Corporate wide area networks, connecting offices within the same national or international company, sit somewhere between. Sometimes these corporate networks use dedicated lines for security, but more often now use virtual private networks (VPNs) layered on top of the Internet.

Mobile phones have been placed within the global-fixed category as well. This may seem strange—the phone can go anywhere. However, the interconnections between phones are fixed and location independent. If two mobile phones are in the same room, it is no easier to connect between them than if they are at opposite ends of the earth (bar a shorter lag time perhaps).

Similarly, the Internet although increasingly accessible through mobile devices and phones is largely based on fixed domain names, Internet Protocol (IP) numbers, and URLs.

Given the placement of mobile phones is a little ambiguous, and it is possible to detect the location of phones and thus deliver location-based content, some of the phone technologies have been listed in the global-flexible category: Global System for Mobile Communications (GSM), General Packet Radio Service (GPRS), and 3G. There is obviously a steadily increasing data rate and third-generation services are able to cope with heavy media content including live video. A few years ago, it seemed like the “killer app” to sell these services would be live sports highlights, but actually it is users’ viewing and uploading of their own content: videos to YouTube and photos to Flickr or Facebook that are the main use of this bandwidth. Perhaps, the most significant changes in mobile telecoms have been in the charging and connectivity model. With GSM you are connected when required to the Internet, and this was treated like any other telephone call, usually meaning pay per minute while connected. In contrast, second- and third-generation services are based on sending small packets of data (the P in the GPRS acronym). The connection to the Internet is treated as “always on” and packets of data are sent to or from the phone as required. Charging is also typically by data use or by fixed charge.

In the local-flexible category, there is a host of existing and emerging technologies. At the most mundane are the now ubiquitous Wi-Fi networks and hotspots (based on the 802.11 Protocol) (IEEE 2001). These merely treat the machine the same as if it were plugged into the local fixed network. At a more local scale, infrared (IrDa)-enabled devices can talk to one another if their infrared sensors are within line-of-sight and Bluetooth (Bluetooth 2001) or emerging wireless technologies such as ZigBee or ultra wideband (Zigbee 2006; WiMedia 2006) allows flexible connections between personal devices. With these a laptop can use a mobile-phone modem, or a Bluetooth hands-free headset can connect to a phone without having to plug in with a piece of wire.

These same technologies can also be used to establish local connections with printers or other devices or even track people using the unique addresses that are often broadcast continually. Thus they offer both the opportunities of accessing fixed public equipment through personal devices, but also the threat of surveillance and hacking everywhere!

Also operating at very close range are various forms of near-field communication technology and radio-frequency identification tags. These are often operated by very close contact, such as touching a train pass to a reader, or wheeling a shipping trolley full of tagged goods near a reader.

Finally, research in wearable computers has suggested using the body itself as the connection between worn devices in a personal area network (Zimmerman 1996). The future is networked and we will become the network.

On the whole, we have seen in the last 10 years the main focus of network-based interaction has moved anticlockwise in this picture from fixed/local networks (mainly LAN), through fixed global networks (the Internet and web explosion), through global mobile networks (mostly phone-based, but including Wireless Application Protocol [WAP], i-mode, etc.) and moving toward flexible local connections between devices. In both the local and global spaces, there has also been a growth of less centrally-controlled networking with peer–peer services establishing decentralized applications over the Internet and wireless ad hoc networks allowing machines to establish networks with no fixed infrastructure.

11.3 NETWORKS AS ENABLERS: THINGS THAT ARE ONLY POSSIBLE WITH NETWORKS

It can be the case that the network is no more than an implementation detail—for example, using a networked disk rather than a local one. However, there are also many applications, like videoconferencing, which are only possible because the network is there. The key feature of networks is the access to remote resources of some kind or other.

11.3.1 Remote Resources

Four kinds of remote things are made accessible by networks:

- People
- Physical things
- Data
- Computation

These may be remote because they are far away from where you normally are, or because you are yourself on the move and hence away from one's own resources (colleagues, databases etc.). Thus mobility can create a need for any or all the above.

11.3.1.1 People

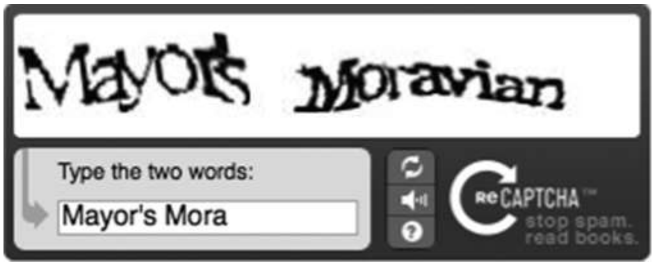
Networks mean we can communicate and work with others in distant places. This is often a direct action, such as e-mailing someone or engaging in a videoconference. These are all the normal study of computer-supported cooperative work (CSCW) and groupware (see Chapter 29).

Interaction with remote people may also be indirect. Recommender systems gather information about people's preferences and use this to suggest further information, services, or goods based on their own preferences and those of others who have similar tastes (Resnick and Varian 1997; Konstan 2004; Riedl and Dourish 2005). Because the people making recommendations are in different locations from each other, the data on who selected what must be stored centrally, or at least with some central control. If you have been suggested books at Amazon, you have experienced a recommender system.

Collaborative virtual environments, such as Second Life, also offer the ability for remote people to interact, but by embedding them within an apparently local virtual reality world. Although the people you are dealing with may be half a world away, their avatar (a virtual presence, perhaps a cartoon character, photo, or robot-like creature) may seem only a few yards or meters away in the virtual world.

Networking has made remote working possible for many years both telecommuting from home and also more nomadic teleworkers such as sales representatives on the road or in hotels (Denbigh 2003). These are largely traditional working relationships, simply freed from the constraints of the office desk. However, networks have made possible a number of more radical remote working styles. Amazon's Mechanical Turk (Amazon 2010) and similar marketplaces allow small pieces of work to be requested, executed, and paid where the person wanting the work and the person doing it have no contact or knowledge of each other except through the website.

More radical are various forms of human computation (also known as crowd sourcing), where substantial tasks are achieved through the small actions of many people, often in the form of a game or puzzle. The most well-known of these is reCaptcha, which is used as a way to ensure users of a web page or service are human and not an automated agent (von Ahn et al. 2008). reCaptcha shows slightly distorted words, which the user needs to type in correctly in order to proceed in the site (see Figure 11.1). However, unlike many schemes where the words are algorithmically distorted from known text, in reCaptcha, the text displayed comes from documents where optical character recognition (OCR) has failed. One of the words is known, but the other is unknown, so that the user is effectively reading the unrecognized word and so slowly increasing the corpus of known text.



Type the two words:

Mayor's Mora

FIGURE 11.1 reCaptcha interface (usually embedded in a web page).

Technology often favors those who are already more materially well off, disadvantaging the poor, the old, and those in countries with a less well-developed technical infrastructure. However, there are some systems which counter this trend.

One is the Net Neighbors scheme in York (Blythe and Monk 2005). Many supermarkets have Internet-based shipping services delivering directly to the home. This would be a great benefit to the elderly, especially those who are house bound or with limited mobility, but these are precisely those least likely to have access to the Internet or be able to use it. Net Neighbors pairs an elderly client with a volunteer. The client telephones the volunteer and dictates a shopping list. The helper then goes to the supermarket site and fills in the online order for delivery direct to the client's doorstep.

Possibly more revolutionary is txteagle (Eagle 2009; txteagle 2009). Txteagle is rather like Mechanical Turk, sending small tasks to independent workers who have registered willingness to perform small tasks. However, the workers in this case are largely illiterate and the tasks (e.g., translation of short phrases) are delivered through mobile phones.

11.3.1.2 Physical Things

We can also view and control remote things at a distance. For example, live webcams in public places allow us to see things (and people) there. Similarly the cameras mounted around rockets as they prepare to take off (and then usually destroyed during the launch) allow the mission controllers to monitor critical aspects of the physical system as do the numerous telemetry sensors, which will also be related through some sort of closed network. And of course the launch command itself will be relayed to the rocket by the same closed network as will the ongoing mission, perhaps the Mars robots, through wireless links.

In the rocket example, it would be dangerous to be in the actual location; in other circumstances, it is merely expensive or inconvenient. Telescopes are frequently mounted in distant parts of the world where skies are clearer than those above the laboratories to which they belong. In order to avoid long international trips to remote places, some of these now have some form of remote control and monitoring using the Internet (Lavery, Kilgourz, and Sykeso 1994).

At a more personal level, the systems within certain high-end cars are controlled using a within car network (called controller-area network (AN)). Even an adjustable heated seat may require dozens of control wires, but with a network, only one power and one control cable is needed. The engine management system, lighting assemblies, radio, CD player, windscreen wipers, each have a small controller that talks through the network to the drivers console (although critical engine systems will usually have a separate circuit).

Many household appliances are now being made Internet-ready. In some cases, this may mean an actual interface—for example, an Internet fridge that can scan the bar-codes of items as you put them in and out and then warn you when items are getting out of date, generate a shopping list of items for you, and even order from your favorite store (Electrolux 1999). Others have instead, or in addition, connectivity for maintenance purposes, sending usage and diagnostic data back to the manufacturer so that they can organize service or repair visits before the appliance fails in some way.

Although these appliances have been available for some years, they are still extremely rare as the benefits for end users rarely seem to justify the substantial price tags. This may change as the environment around becomes increasingly networked. For example, large numbers of consumer items are seen expected to be tagged using near-field communication. The purpose of this is to help stock control and speed up checkouts in shops. However, this will mean that an intelligent refrigerator will be able to identify its contents, know when they were purchased, by-passing the bar-code scanning of first-generation Internet refrigerators.

Another example, that is already with us is the use of iPhones in the home to control (suitable enabled) Hi-Fi equipment, or to remotely program satellite TV receivers. The latter is interesting as the TV receivers already have telephone connections to download program information, so are already "networked" and the iPhone also has semipermanent network connectivity; furthermore the benefit, being able to decide to record a program when not at home, is substantial. We are beginning to see signs of the synergy possible between networked devices.

In some ways, Internet shopping can also be seen in this light. While at one level it is merely a transfer of data, the ultimate end is that you receive physically the ordered goods. This interaction with the remote physical goods is often two-way as you track its progress, and sometimes its physical location through a web interface.

Perhaps most important in a world facing global warming is the potential for the additional information available through networks to improve energy usage. Smart network technologies are already routinely used for very fine grain monitoring of industrial processes and utilities (mqtt.org 2009), and in the home, the smart refrigerator may not only detect when your food is out of date, but also more intelligently monitor power use, making use of periods of low-demand (Anslow 2009). In the past, utilities have managed power production to match consumption, but as renewable energies such as wind-power make power production less controllable, transferring this information down to devices makes managed consumption possible.

11.3.1.3 Data

Anyone using the web is accessing remote data. Sometimes, data is stored remotely purely for convenience, but often data is necessarily stored remotely because

- It is shared by many remote people.
- Central storage helps maintain control, security, or privacy.
- It is used by a single user at different locations (web e-mail).
- It is too extensive to be stored locally (e.g., large databases and thin client).

In the case of the web, the data is remote because it is accessed by different people at different locations, the author(s) of the material, and all those who want to read it.

Even the web is quite complex: We may perceive a web page as a single entity, but in fact it exists in many forms (Figure 11.2). The author of the page will typically have created it offline on his or her own PC. They then upload the page (which effectively means copying it) onto the web server. Any changes the author makes after uploading the page will not be visible to the world until it is next uploaded. When you want to see the page and enter a URL or click on a link, the browser asks the web server for the file which is then copied into the browser's memory and displayed to the user. You can tell the browser has a copy as you can disconnect from the Internet and still scroll within the file. If you access the same page again quite soon, your browser may choose to use the copy it holds rather than going back to the web server, again potentially meaning you see a slightly out-of-date copy of the page. Various other things may keep their own cached copies including web proxies and firewalls.

This story of copied data in various places is not just about the web, but true to some extent or other of all shared networked data. With people or physical things, we do not expect to have the actual person or thing locally, just a representation. This is equally true for shared data, except that the representation is so much like the "real thing;" it is far less obvious to the user.

For shared networked data even the "real thing" may be problematic—there may be no single "golden copy," but instead many variants all with equal right to be called "the real data."

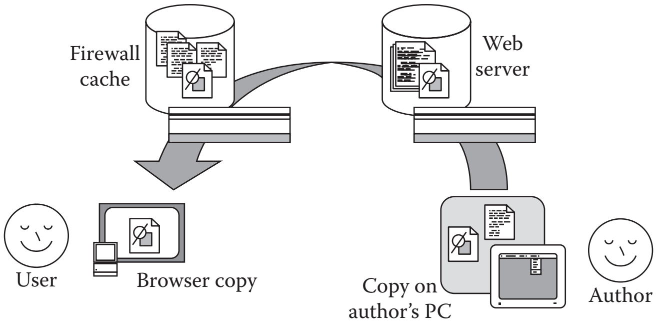
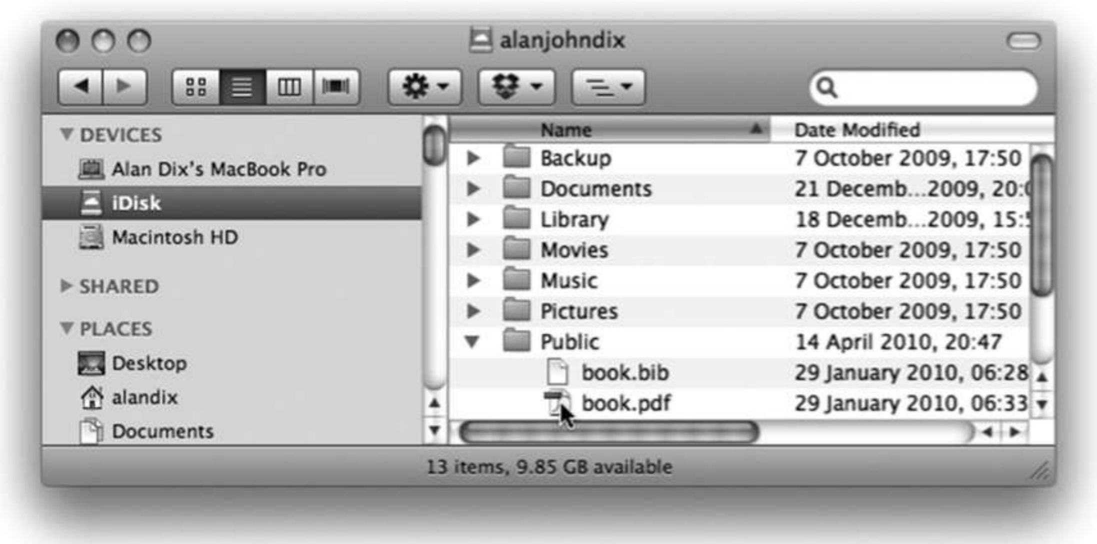


FIGURE 11.2 Copies of a web page in many places.



Name	Date Modified
Backup	7 October 2009, 17:50
Documents	21 Decemb...2009, 20:(
Library	18 Decemb...2009, 15:5
Movies	7 October 2009, 17:50
Music	7 October 2009, 17:50
Pictures	7 October 2009, 17:50
Public	14 April 2010, 20:47
book.bib	29 January 2010, 06:28
book.pdf	29 January 2010, 06:33

13 items, 9.85 GB available

FIGURE 11.3 Apple iDisk—cloud data on the desktop.

You do not even escape networking issues if you only access data locally on your own PC—networking issues may still arise if your data is backed-up over the network, and with cloud data services, virtual disks on the desktop may in fact be simply links to a network that is distributed across the world. For example, in MacOSX, an iDisk is stored remotely on Apple servers, but appears similar to an ordinary folder or local disk (Figure 11.3).

As well as having multiple copies of data, distributed sources and computation can sometimes make it hard for the user to know what the data actually is. For example, when a web page is generated at the server, there may be no "page" at all until data is assembled from a database or web sources at the point of creation, and for a web page created dynamically from AJAX sources, the page delivered, as in Figure 11.2, may be very different than the page viewed by the user.

11.3.1.4 Computation

Sometimes it is remote computational resources that are accessed over the network. The most obvious example of this is large supercomputers. These have enormous computational power, and scientists wishing to use them will often prebook time slots to perform particularly intensive calculations such as global weather simulations, analysis of chemical structure, stress calculations, and so on. Because these machines are so expensive, programs for them are typically developed on other less powerful computers and then uploaded over the network when the supercomputer is available.

If the data required as input or output for the calculation is not too great, "fairly simple" means can be used to upload the programs and data. However, some calculations work on large volumes of data—for example, data from microwave readings of the upper atmosphere to probe the ozone hole generate terabytes (millions of millions of bytes) of data per second. High-capacity networks are being created in many countries to enable both high-volume data for this sort of application and also the expected data required for rich media (Foster 2000; Foster and Kesselman 1999; GRID 2001).

The ease with which data and results can be shipped back and forth across the Internet has enabled the growth of web services: web applications designed to be accessed by other programs supplying services or data. In addition to more scientific or heavy commercial uses of these, they have become a standard part of many consumer-oriented applications, for example, del.icio.us has a standard API accessible through the web allowing third-party applications to interact with it.

Sometimes calculations need to be performed centrally, not because the central computer is powerful, but because the local device is a computational lightweight. For example, one may want to create a remote analysis package where engineers in the field enter data into a personal digital assistant (PDA) or phone interface, but where complex stress calculations are carried out on a small server back in the office. The data on materials and calculations involved may not be extensive by supercomputer standards, but may still be too much for a handheld device.

Because transporting large volumes of data is not always practical, calculations are often performed where the data is. (In performing any computation program, data and computational engine must all be in the same place. If they are not together, then one or the other must be moved or copied to bring them together [Ramduny and Dix 1997].) For example, when you perform a database access, the request for the data is usually transmitted to the database server as an SQL query, for example, "SELECT name, salary FROM payroll WHERE salary > 70000." In principle, the complete contents of the payroll database could be downloaded to your PC and the selection of appropriate records carried out locally; however, it would be more costly to transmit the data, hence the calculation is effectively transmitted to the database server. In a similar vein, Alexa allow third parties to run programs on their servers through their Web Search Platform (AlexaWSP 2006); this allows the programs to access a 100-TB web crawl that would be impractical (and commercially unacceptable) to transfer to clients.

Even when the volume of data is not large or the frequency of access would make it cost effective to transmit it, security or privacy reasons may prevent the download of data. For example, some datasets are available to search to a limited degree on the web, but charge for a download or CD of the complete dataset. My own http://hcibook.com site (Dix et al. 1998) is rather like this, allowing searching of the book's contents online and displaying portions of the text, but not allowing a full download, as readers are expected to buy the book!

Security considerations may also prohibit the distribution of programs themselves if they contain proprietary algorithms. Also if the source of the program is not fully trusted, one may not want to run these programs locally. The latter is the reason that Java applets are run in a software "sandbox" confining the ability of the applet to access local files and other potentially vulnerable resources.

For those who have not come across it, the Search for Extra-Terrestrial Intelligence (SETI) project is analyzing radio signals from outer space looking for patterns or regularities that may indicate transmissions from an alien civilization. You can download a SETI screensaver that performs calculations for SETI when you are not using your machine. Each SETI screensaver periodically gets bits of data to analyze from the central SETI servers and then returns results. This means that the SETI project ends up with the combined computational resources of many hundreds of thousands of PCs.

SETI is an interesting example of remote computation (SETI@home 2001). Normally remote computation involves a device of low-computational power asking a central computer to do work for it. In the case of SETI, large calculations are split up and distributed over large numbers of not particularly powerful computers.

The same technique is used in "PC farms." These are when large numbers of PCs are networked together to act as a form of supercomputer. For example, in CERN (the home of the web), data from high-energy collisions may consist of many megabytes of data for each event, with perhaps hundreds of significant events per second (CERN 2001). The data from each event is passed to a different PC, which then performs calculations on the data. When the PC finishes, it stores its results and then adds itself back to a pool of available machines.

During coming years, we are likely to see both forms of remote computation. As devices become smaller and more numerous, many will become simply sensors or actuators communicating with central computational and data servers (although central here may mean one per room, or even one per body). However, inspired by SETI, several companies are pursuing commercial ways of harnessing the spare, and usually wasted, computational power of the millions of home and office PCs across the world.

If the network is fast enough, it no longer matters where computation happens, hence the growth of cloud computing turning processing into a commodity—if you have a certain amount of computation to do, instead of buying a dedicated machine, you pay for it to happen "somewhere." However, the exception is for highly time-critical tasks and in particular rapid feedback from user interaction. We shall discuss this more in Section 11.4.2, but the crucial thing is that for many interactions, response times of less than a second or in some cases closer to 100 milliseconds are required; hence the move for highly interactive websites to using JavaScript running in the browser rather than server-side computation.

11.3.2 Applications

The existence of networks, particularly the global networks offered by the Internet and mobile-phone networks, have made many new applications possible and changed others.

Several of the more major application areas made possible by networks are discussed in their own chapters: groupware (Chapter 29), online communities (Chapter 30), mobile systems (Chapter 32), e-commerce (Chapter 39), telecommunications (Chapter 40), and of course, the web (Chapter 37).

In addition, networking impinges on many other areas. Handheld devices (Chapter 32) can operate alone, but are increasingly able to interact with one another and with fixed networks through wireless networking. Similarly wearable computers (Chapter 33) are expected to be interacting with one another through short-range networks, possibly carried through our own bodies (makes mobile phones seem positively safe!) and information appliances (Chapter 38) will be Internet connected to allow remote control and maintenance. In the area of government and citizenship (Chapter 41), terms such as e-democracy and e-government are used to denote not just the technological ability to vote or access traditional government publications online, but a broader agenda whereby citizens feel a more intimate connection to the democratic process. Of course, education, entertainment, and game playing are also making use of networks.

Throughout the chapter, we will also encounter broader issues of human abilities, especially concerned with time and delays, involving aspects of virtually all of Part II (human perception, cognition, motor skills, etc.). Also we will find networking raises issues of trust and ethics (Chapters 62 and 65), and of course, the global network increases the importance of culturally and linguistically accessible information and interfaces (Chapter 23).

Networking has already transformed many people's working lives allowing telecommuting, improving access to corporate information while on the move and enabling the formation of virtual organizations. Networks are also allowing whole new business areas to develop, not just the obvious applications in e-shopping and those concerned with web-design.

The Internet has forced many organizations to create parallel structures to handle the more direct connections between primary supplier and consumer (disintermediation).

This paradoxically is allowing more personalized (if not personal) services and often a focus on customer–supplier and customer–customer communication (Siegal 1999; Light and Wakeman 2001). This restructuring may also allow the more flexible businesses to revolutionize their high street (or mall) presence—allowing you to buy shoes in different sizes, or next day fitting services for clothes (Dix 2001b).

The complexity of installing software and the need to have data available anywhere at any time has driven the application service provider sector. You do not install software yourself, but use software hosted remotely by providers who charge on a usage rather than once-off basis. By storing the data with third parties, an organization can off-load the majority of its backup and disaster management requirements.

This process has accelerated with the growth of cloud computing. Providers of these services store and deliver data anywhere in the world using networks of servers, but without it being apparent where on the net the data is stored. More critical computation is available at one of many servers, again without the client needing to be concerned where and when the computation is occurring. For web-based services, this has allowed far more scalable applications, as the provider of the cloud platform will have capacity to allow for a sudden surge in activity. Furthermore, the fact that these services are often charged on a usage basis significantly reduces the barriers to entry for new businesses.

For the individual user, the ubiquity of Internet access for many has enabled many personal information management applications such as e-mail, calendars, bookmark lists, and address books. These things that would once have been seen as personal are being not only accessed through the web, but in many cases also shared. These web communities are no longer the province of geeks, but have become part of day-to-day life of many engendering whole new ways of finding out and getting to know including social bookmarking, blogs, photoblogs (or photologs or phlogs) and social networking sites. New issues arise as personal data becomes distributed over different websites. Who owns it? How do you know what is there? Can you take it with you between sites? The DataPortability Project, is one initiative in this area attempting to create standards and influence industry attitudes so that it is easier to move personal data between sites (DataPortability 2010).

The "personal" device has not become redundant, though, in this web-orientation of applications. In addition to being an access point to global services, it is also a potential interaction device for things close by. For example, in an installation by .:thePooch:. (thePooch 2006) in an arts event the attendees were encouraged to send SMS texts to Andrine, a huge face projected high on the wall. The texts were analyzed using natural language-processing techniques and depending on the content the face took on different emotions: happy, sad, shocked (Lock, Rayson, and Allanson 2003). The cameras in phones are also being used to enable them to be used as location-finding devices (Sarvas et al. 2004), to enable the embedding of SpotCodes or other visual codes in paper posters (Toye et al. 2004; Semacode 2006) and for real-time manipulation of large public displays (Rohs, Sheridan, and Ballagas 2004). Some of these applications use local networking such as Bluetooth, others paradoxically use the "global" connectivity through SMS or WAP to enable local interactions.

11.3.3 Virtual Networks

The word "network" has two meanings: There is the technical communication infrastructure of wired and wireless connections between machines, but atop this various virtual networks of web pages, social networks, and linked data.

The web itself is a network in the sense that each page links to others, creating a network of connections. Of course, it is possible to imagine such networks without there being an underlying communications network. Indeed this was precisely the nature of most early hypertext systems, such as Notecards, which ran on early Xerox Lisp machines (Halasz et al. 1986). However, the fact that the web sits on top of the Internet means it can have a scale beyond anything one would normally consider in a single machine. In fact, this scale is not so much the information capacity; the crawled web is less than a petabyte and can fit in a rack of disks the size of a filing cabinet (Dix 2005). The crucial thing is that a distributed web can be updated in many places my many people, it is the underlying network of people who are connected through the Internet that makes the knowledge embodied in the web possible.

When in 2005 Tim O'Reilly wrote down some of the characteristics of the then emerging web2.0 (O'Reilly 2005), one of the key aspects was "Harnessing Collective Intelligence." In some ways, this creative power of distributed people was already evident in even the earliest web. However, in second-generation web systems such as Wikipedia this became far more clear, as the distributed collective knowledge was brought together. That is, it is the power of the communications network to connect people combined with its power to connect data that makes both the original web and the second generation web2.0 possible.

This interconnecting of people is at the heart of the social networks, which are now part of so many people's lives. Sociologists study the networks of connections between people where there are no computers or telecommunications. However, it is clear that social network sites such as Facebook, Hi5, and MySpace, and microblogging such as Twitter have created something new. Sometimes this is merely another mode of communication for existing nonweb interactions and some social network sites, such as Facebook, are oriented primarily toward this reinforcement of existing connections. Others are more oriented toward forging new connections.

Computer readable data is at the centre of the semantic web (Berners-Lee et al. 2001), which seeks to add explicit semantics to existing web resources … and ultimately everything! The semantic web centers on a number of key technologies. One of the most important is resource description framework (RDF), which is effectively a way of discussing the relationships between "resources," where a "resource" is anything for which you can create a uniform resource identifier (URI). This was initially web pages or parts of them, but over time URIs were defined for nonweb resources such as books, places, and people. This graph of resources and relationships forms another network, but in early semantic web, applications were largely about the web, but each graph of RDF existed separately.

However, the emerging “web of data” or “linked data” is harnessing the communications infrastructure of the web to allow distributed yet interconnected RDF data (Bizer, Heath, and Berners-Lee 2009). The key idea that makes this possible has been to make URIs of nonweb resources actually be dereferenceable—that is, being web documents that describe the resource. For example, the URI http://www.alandix.com/rdf/alandix_xml links to a FOAF (friend-of-a-friend) file (FOAF 2000), which describes some facts about me in RDF (see Figure 11.4). Because of this, the web of knowledge in the linked data world can bridge between individual repositories, if a URI is encountered, it can be looked up to find more information about it, and from it further resources can be accessed. Figure 11.5 shows a map of linked data available in July 2009 including Dbpedia, Wikipedia data represented in RDF. However, since this time, the U.K. government has released vast amounts of data at http://data.gov.uk and other governments around the world are doing the same.

For the user interface, this creates new challenges; how to represent this machine-formatted data effectively to the user? However, there are also enormous opportunities, as massive amounts of data are available to augment user interactions (Dix et al. 2010).

11.4 NETWORKS AS MEDIATORS: ISSUES AND PROBLEMS BECAUSE OF NETWORKS

This section takes as a starting point that an application is networked and looks at the implications this has for the user interface. This is most apparent in terms of timing problems of various kinds. This section is really about when the network is largely not apparent except for the unintended effects it has on the user.

We will begin with a technical introduction to basic properties of networks and then see how these affect the user interface and media delivery.

11.4.1 Network Properties

11.4.1.1 Bandwidth and Compression

The most commonly cited network property is bandwidth—how much data can be sent per second. Those who have used dial-up connections will be familiar with 56-K modems, and those with better memory or those using mobile-phone modems may recall 9.6-K modems or less. The “K” in all of these refers to thousands of bits (0/1 value) per second (strictly Kbps) rather than bytes (single character) that are more commonly seen in disk and other memory sizes. A byte takes 8 bits, and considering a small amount for overhead, you can divide the bits per second by 10 to get bytes per second.

Faster networks between machines in offices are more typically measured in megabits per second (again strictly Mbps but often just written M)—for example, the small “telephone cable” Ethernet is rated at either 10 Mbps or 100 Mbps.

As numbers these do not mean much, but if we think about them in relation to real data, the implications for users become apparent.

A small word processor document may be 30 Kb (kilo bytes). With a 9.6-K GSM modem, this will take approximately half a minute, on a 56-K modem this is reduced to 5 seconds, for a 10-Mb Ethernet this is 30 milliseconds. A full screen web quality graphic may be 300 Kb taking 5 minutes of 9.6-K modem, less than a minute on a 56-K modem, or

```
<?xml version =“1.0” encoding=“UTF-8”?
<rdf:RDF    …>
  <foaf:PersonalProfileDocument rdf:about =“http://www.alandix.com/rdf/alandix.xml”>
  <foaf:maker rdf:resource =“#me”/>
  <foaf:primaryTopic rdf:resource =“#me”/>
  <foaf:Person rdf:ID =“me”>
    <foaf:name>Alan Dix</foaf:name>
    <foaf:title>Prof</foaf:title>
    <foaf:givenname>Alan</foaf:givenname>
    <foaf:family_name>Dix</foaf:family_name>
    <foaf:mbox_sha1sum>…</foaf:mbox_sha1sum>
    <foaf:homepage rdf:resource =“http://www.alandix.com/”/>
    <foaf:depiction rdf:resource =“http://www.alandix.com/images/alan-australia.jpg”/>
    <foaf:knows>
      <foaf:Person>
        <foaf:name>Nadeem Shabir</foaf:name>
        <foaf:mbox_sha1sum>…</foaf:mbox_sha1sum>
      </foaf:Person>
    </foaf:knows>
  </foaf:Person>
</rdf:RDF>
```

FIGURE 11.4 Fragment of resource description framework at http://www.alandix.com/rdf/alandix.xml.

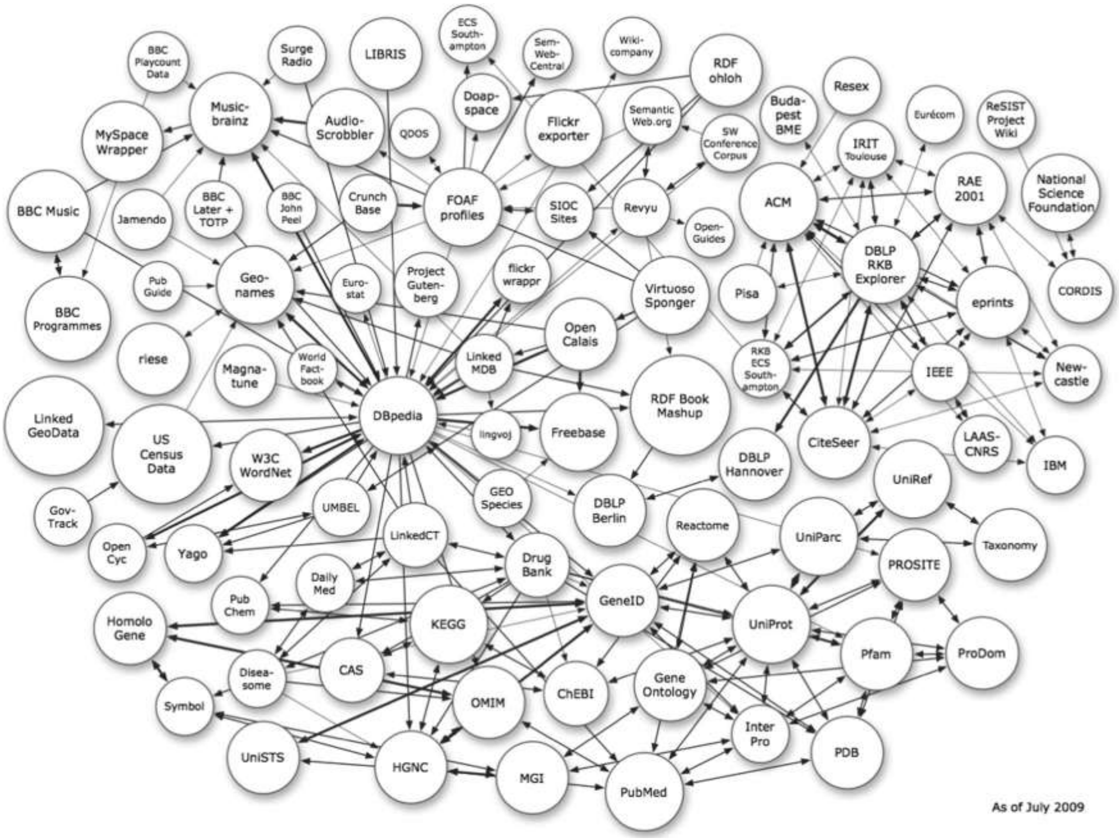


FIGURE 11.5 Linked data as of July 2009 (http://richard.cyganiak.de/2007/10/lod/).

Note that I am using the formula:

$$\text{download time } T = \frac{F \times 10}{M}$$

where F is the size of file in bytes, 10 is the number of raw bits per byte, and M is the modem speed in bits per second.

1/3 second on a 10-Mb Ethernet. (N.B. these are theoretical minimum times if there is nothing else using the network.)

Rich media such as sound or video put a greater load again. Raw, uncompressed Hi-Fi quality sound needs more than 200 Kbps and video tens of Mbps. Happily there are ways to reduce this, otherwise digital AV would be impossible over normal networks.

Real media data has a lot of redundant information—areas of similar color in a picture, successive frames in a video are similar, sustained notes in music. Compression techniques use this similarity to reduce the actual amount of data that needs to be sent (e.g., rather than sending a whole new frame of video, just send the differences from the last frame). Also some forms of compression make use of human perceptual limits: For example, MP3 stores certain pitch ranges with greater fidelity than others as the human ear's sensitivity is different at different pitches (MPEG 2001), also JPEG images give less emphasis to accurate color hue than the darkness/lightness (JPEG 2001). Between them these techniques can reduce the amount of information that needs to be transferred significantly, especially for richer media such as video. Thus the actual bandwidth and the effective bandwidth, in terms of the sorts of data that are transmitted may be very different.

11.4.1.2 Latency and Start-Up

Bandwidth measures how much data can be transferred—latency is how long each bit takes (see Figure 11.6). In terms of a highway, bandwidth would be how many lanes and latency is the time it takes to travel the length of the highway. The latency is due to two factors. The first is the speed of transmission of electricity through wires or light through optical networks. This may seem insignificant, but for a beam of light to travel across the Atlantic would take 20 milliseconds and in practice this hop takes more like 70 milliseconds. For satellite-based communications, the return trip to and from a geostationary satellite takes nearly a second; think about the typical delay you can hear on a transcontinental telephone call. The second factor contributing to latency is that every electronic switch or computer router has to temporarily store, and then decide what to do with the signal before passing it on to the next along the chain. Typically, this is a more major factor, and in practice, trans-Atlantic Internet traffic will take nearly 250 milliseconds from source to final destination, most of which in various computer centers at one end or the other.

Latency is made worse by set-up time. Every time you establish an Internet connection, a conversation is established between your computer and the machine hosting the web server:

> "hello are you there,"
> "yes I'm here what do you want,"
> "I'd like to send you some data,"
> "great I'm waiting,"
> "OK here it is then" (this is called handshaking).

Each turn in this conversation involves a round trip, network latency on both outward and return paths and processing by both computers. And this is before the web server proper even gets to look at your request. Similar patterns happen as you dial a telephone call.

Latency and set-up time are critical as they often dominate the delay for the user except for very large files or streaming audio/visual media. Early web design advice (by those concerned about people with slow connections, but who clearly had never used one!) used to suggest having only as much text that would fit on a single screen. This was intended to minimize the download time. However, this ignores set-up times. A long text page does not take long to load even on a slow connection, once the connection to the web server has been established. Then it is far faster to scroll in the browser than to click and wait for another small page to load. A similar problem is the practice of breaking large images up into a jigsaw of small pieces. There are valid reasons for this—allowing rollover interaction or where parts of the image are of different kinds (picture/text)—however, it is also used without such reasons and each small image requires a separate interaction with the server encountering latency and set-up delays.

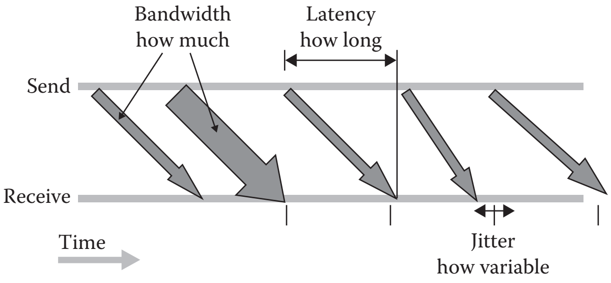


FIGURE 11.6 Bandwidth, latency, and jitter.

11.4.1.3 Jitter and Buffering

Suppose you send letters to a friend every 3 days and the postal service typically takes 2 days to deliver letters (the average latency in network terms). Your friend will receive letters every 3 days, just delayed from when you sent them. Now imagine that the postal system is a little variable, sometimes letters take 2 days, but occasionally they are faster and arrive the next day, and sometimes they are slower and take 3 days. You continue to send letters every 3 days, but if a slow letter is followed by a fast one, your friend will receive them only one day apart, if on the other hand a fast letter is followed by a slow one, the gap becomes 5 days. This variability in the delay is called jitter. (Note that the fast letters are just as problematic as the slow ones—a fast letter followed by a normal speed one still gives a 4-day gap.)

Jitter does not really matter when sending large amounts of data, or when sending one-off messages. However, it is critical for continuous media. If you just played video frames or sound when it arrived, jitter would mean that the recording would keep accelerating and slowing down (see Figure 11.7a and b).

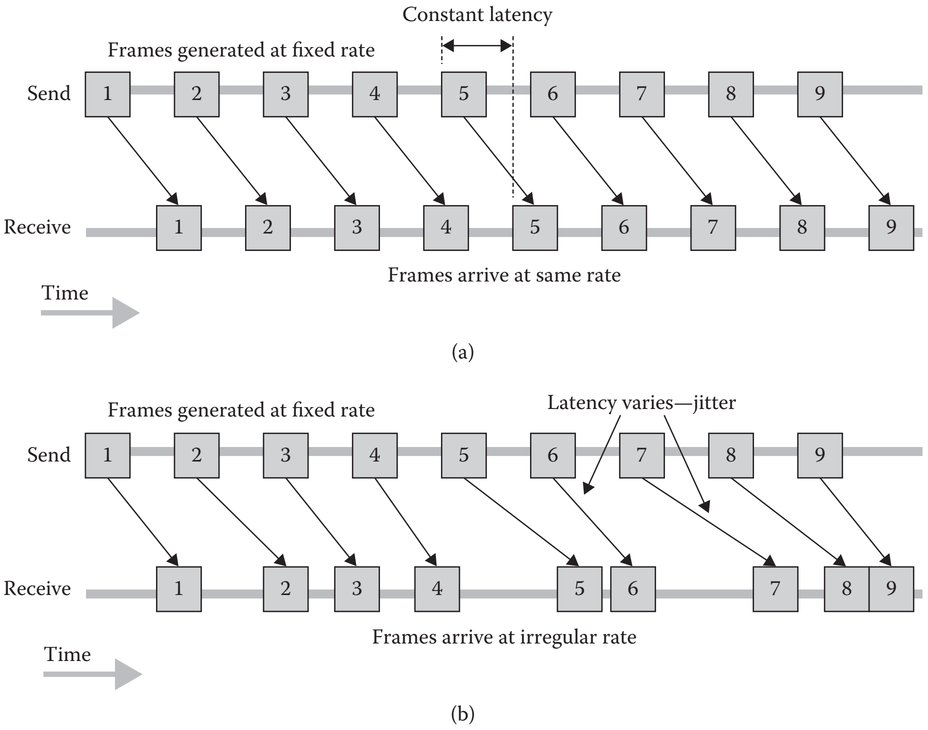


FIGURE 11.7 (a) No jitter—no problem. (b) Jitter causes irregular reception.

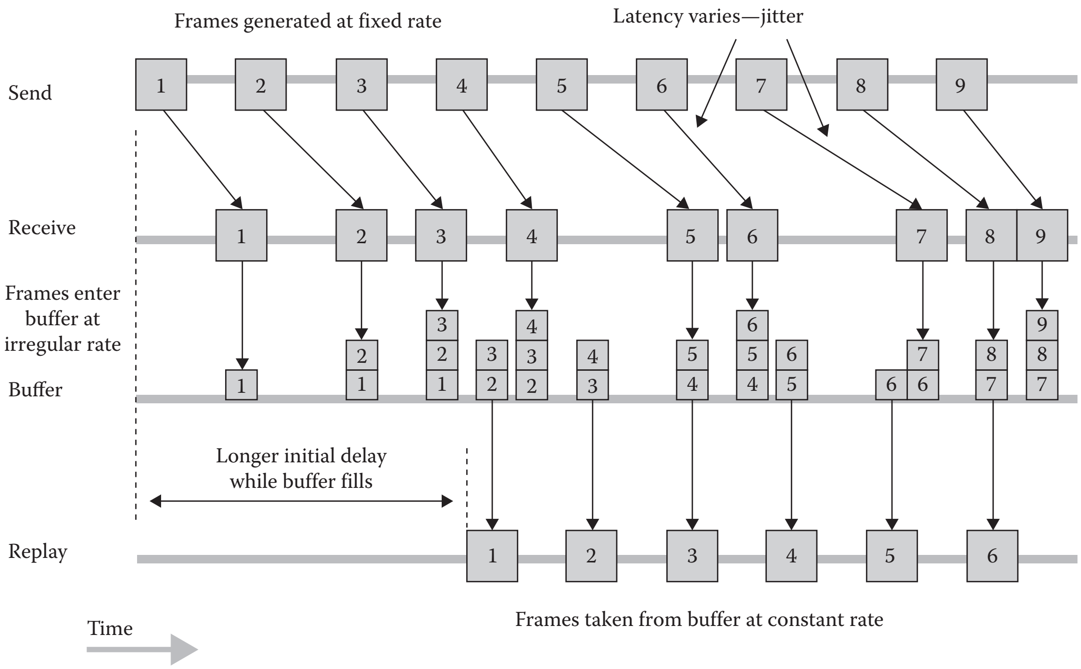

Frames generated at fixed rate
Latency varies—jitter
Send
Receive
Frames enter buffer at irregular rate
Buffer
Longer initial delay while buffer fills
Replay
Time
Frames taken from buffer at constant rate

FIGURE 11.8 Buffering will smooth the jitter, but adds delay.

Jitter can be partially alleviated by buffering. Imagine your friend's postman holds back one letter for 3 days and then starts giving letters to your friend one every third day. If your mail always arrives in exactly 2 days, the postman will always hold exactly one letter as mail will arrive as fast as he passes it on. If however a letter arrives quickly, he will simply hold 2 letters for a few days and if it is slow, he will have a spare letter to give. Your friend's mail is now arriving at a regular rate, but the delay has increased to (a predictable) 5 days. Buffering in network multimedia behaves exactly the same holding back a few seconds audio/video data and then releasing it at a constant rate (see Figure 11.8).

11.4.1.4 Reliability and Loss, Datagram and Connection-Based Services

Virtually all networks are designed on the principle that there will be some loss or damage to data en-route. This arises for various reasons—sometimes there is electrical interference in a wire, and sometimes the internal computers and routers in the network may have too much traffic to cope with. This is normal and network software is built to detect damaged data and cope with lost data.

Because of this, the lowest layers of a network are assumed to be lousy. Any data damaged in transit are discarded, and computers and hardware en-route can choose to discard data if they get busy. So when one computer sends a packet of data to another, it can assume that if the packet of data arrives it will be intact, but it may never arrive at all.

Some network data, in particular certain forms of real-time multimedia data are deliberately sent in this unreliable, message-at-a-time, form (called datagrams).

However, it is usually easier to deal with reliable channels, so higher levels of the network create what are called connection-based services on top of the unreliable lower level service. Internet users may have come across the term Transmission Control Protocol (TCP)/IP. IP is the name of an unreliable low-level service that routes packets of data between computers. TCP is a higher level connection-based service built on top of an IP. The way TCP works is that the computer wanting to make a connection contacts the other and they exchange a few (unreliable IP) messages to establish the link. Once the link is established, the sending computer tags messages it sends with sequence data. Whenever the receiving computer has all the data up to a certain point, it sends an acknowledgement. If the sending computer does not get an acknowledgement after a certain time, it resends the data (Stevens 1998, 1999).

With TCP, the receiving computer cannot send a message back, and when it notices a gap, it has to wait for the sending computer to resend after the timeout. While it is awaiting the resend, it cannot process any of the later data. Notice that this means reliability is bought at the price of potential delays.

11.4.1.5 Quality-of-Service and Reservation

The above properties are not just determined by a raw network's characteristics, such as the length of wires, types of routers, modems, and so on. They are also affected by other traffic—its volume and nature. If 10 PCs are connected to a single 10-Mbps network connection and require high-volume data transfers (perhaps streaming video), then there is only, on average, 1 Mbps available for each. If you are accessing a network service that requires trans-Atlantic connections during peak hours, then intermediate routers and hubs in the network are likely to become occasionally overloaded leading to intermittent packet loss, longer average delays, and more variability in delay, hence jitter. In the past, when the capacity of Internet backbones was lower, it was very obvious in the United Kingdom when the United States "woke up" as the web ground to a crawl!

For certain types of activity, in particular real-time or streaming rich-media, one would like to be able to predict or guarantee a minimum bandwidth, maximum delay and jitter, and so on. These are collectively called quality-of-service (QoS) issues (Campbell and Nahrstedt 1997). Some network protocols allow applications to reserve a virtual channel with guaranteed properties; however, the most common large-scale network, the Internet, does not have such guarantees, and it operates solely on a best endeavor basis. Upgrades to the underlying protocol (called by the catchy name IPv6), allow some differentiation of different types of traffic. This may allow routers to make decisions to favor time-critical data, but it will still not be able to reserve guaranteed capacity. However, in practice, the increased capacity of Internet backbones is allowing large-scale Voice-over-Internet services, such as Skype, with acceptable end-to-end service (FCC 2006; Skype 2006).

11.4.1.6 Encryption, Authentication, and Digital Signatures

Some networks, such as closed office networks, offer no greater worries about security of information than talking together (both are capable of being bugged, but with similar levels of difficulty). However, more open networks such as the Internet, or phone networks, mean that data is travelling through a third party and public infrastructure to get to its recipients. Increasing use of wireless devices also means that it is easier for the data sent between devices to be monitored or interfered with by third parties. One option is to only use physically secure networks, but for economic reasons, this is often not an option.

Furthermore, solutions that do not rely on the network itself being secure are more robust. If you rely on, for example, a private dedicated line between two offices and assume it is secure, then if someone does manage to tap into it, all your interoffice communication is at risk.

The more common approach now is to assume the networks are insecure and live with it. This gives rise to two problems:

secrecy—how to stop others from seeing your data
security—how to make sure data is not tampered with

The first problem is managed largely by encryption methods—ensuring that even if someone reads all your communications they cannot understand them (Schneier 1996). The "https" in some URLs is an example of this denoting that the communication to the web server is encrypted.

The second problem, security, has various manifestations. Given communications are through a network, how do you know that you are talking to the right person/machine. Authentication mechanisms deal with this. In various ways, they allow one machine to verify (usually by secret information that can only be known by the true intended party) that it is talking to the right party.

Even if you know that you are talking to the right person/machine, how do you know that the data you receive has not been changed? This is like receiving a signed letter, but, unbeknownst to you, someone has added some lines of text above the signature, although it really comes from the person you think, the message is not as was sent. If data is being encrypted then this may often implicitly solve this problem as any tampered data is uninterpretable by a third party, who therefore cannot alter it in a meaningful way.

If secrecy is not an issue however, encryption is an unnecessary overhead and instead digital signatures generate a small data block that depends on the whole of the message and secret information known to the sender. It is possible for the recipient to verify that the signature block corresponds to the data sent and the person who is supposed to have sent it. One example of this are "signed applets" where the Java code is digitally signed so that you can choose to only run Java programs from trusted parties.

The choice between different forms of secure network is complex. For example, the U.K. National Health Service installed a dedicated network called N3 to every hospital, family doctor, surgery, and pharmacy in order to support a variety of services including an electronic prescription service (EPS). However other countries introducing EPS (e.g., Bulgaria, Gibraltar) are using standard Internet technology such as VPNs. There were clearly reasons behind the U.K. NHS decision, but interviews with pharmacists suggest problems with the N3 network, and the introduction of EPS seems to have been far slower than those countries adopting a more lightweight approach (Griffiths and Dix 2008).

11.4.2 UI Properties

11.4.2.1 Network Transparency

One of the goals of many low-level network systems is to achieve transparency—that is to make it invisible to the user where on the network a particular resource lays. When you access the web, you use the same kind of URL and same kind of interface whether the web server is in Arizona, Australia, or Armenia. I know that when at home I send an e-mail between two machines less than 2 m apart, the message actually goes all the way across the Atlantic and back—but this is only because I have quite a detailed understanding of the computers involved—as a user I press "send mail" on one machine and it arrives near instantaneously on the other. Cloud storage also creates this sense of transparency, giving access to data from anywhere in the world without caring where the data is actually stored.

Although network transparency has many advantages to the user—you do not care about routes through the network, and so on, there are limits to its effectiveness and desirability. Some years ago, I was at a Xerox lab in Welwyn Garden City in the United Kingdom. Randy Trigg was demonstrating some new features of Notecards (an early hypertext system [Halasz, Moran, and Trigg 1987]). The version was still under development and every so often would hit a problem and a LISP debugger window would appear. After using it for a while, it suddenly froze—no debugger window, no error message, just froze. After a few embarrassing seconds, he hit a control key and launched the debugger. A few minutes of frantic scanning through stack dumps, program traces, and so on, and the reason became clear to him. He had demonstrated a feature that he had last used on his workstation at Palo Alto. The feature itself was not at fault, but required an obscure font that he had on his own workstation, but not on the machine there in Welwyn. When Notecards had requested the font, the system might have thrown up an error window, or substituted a similar font. However, in the spirit of true network transparency, the location of the font should not matter. Having failed to find it on the local machine, it proceeded to interrogate machines on the local network to see if they had it, it then proceeded to scan the Xerox U. K. network, and world network. Eventually, if we had waited long enough, it would have been found on Randy's machine in Palo Alto. Network transparency rarely extends to timing!

Transparency has also been critiqued for CSCW purposes (Mariani and Rodden 1991). It may well be very important to users where resources and people are. For mobile computing also, an executive takes a laptop on the plane only to discover that the files needed are residing on a network file server rather than on the machine itself. If the interface hides location how can one predict when and where resources will be available?

In Section 11.5.3, we will discuss recent work where the presence of intermittent connections, limited range, and variable signal strength is being used as a deliberate feature in interfaces.

11.4.2.2 Delays and Time

As is evident, one of the issues that arises again and again when considering networks is time—how long are the delays, how long to transfer data, how variable, and so on. Networking is not the only reason for delays in applications, but is probably one of the most noticeable—the web has often been renamed the "world-wide wait." There is a longstanding literature on time and delays in user interfaces. This is not as extensive as one might think, largely because for a long time, the prevailing perception in the HCI community was that temporal problems would go away (with some exceptions) leading to what I called the "myth of the infinitely fast machine" (Dix 1987).

One of the earlier influential articles was Ben Shneiderman's review of research findings on delays (Shneiderman 1984)—mainly based on command line interfaces. Since then, there have been a number of workshops and special journal issues on issues of time, sparked largely by web delays (Johnson and Gray 1996; Clarke et al. 1997; Howard and Fabre 1998; Hildebrandt, Dix, and Meyer 2004).

There are three main timescales that are problematic for networked user interfaces:

100 milliseconds —Feedback for hand–eye coordination tasks needs to be less than 100–200 milliseconds to feel fluid. This is probably related to the fact that there are delays of this length in our motor-sensory system anyway. For aural feedback, the timescales are slightly tighter again.

1 second—Timescale for apparent cause–effect links such as popping a window after pressing a button. If the response is faster than this, the effect seems "immediate." This is related to a period of about 1 second that the brain regards as "now."

5–10 seconds—Waits longer than these engender annoyance and make it hard to maintain task focus. This may be related to short-term memory decay.

The 100-millisecond time is hard to achieve if the interaction involves even local network traffic. The 1-second time is usually achievable for local networks (and is assumed by X-Windows systems), but more problematic for long haul networks. The 5- to 10-second time is in principle achievable for even the longest transcontinental connections, but when combined with bandwidth, limitations or overload of remote resources may become problematic. This is especially evident on web-based services where the delay between hitting a link and retrieving a page (especially a generated page) may well exceed these limits, even for the page to begin to draw.

The lesson for UI designers is to understand the sort of interaction required and to ensure that parts of the user interface are located appropriately. For example, if close hand–eye coordination is required, it must run locally on the user's own machine—in the case of the web in an applet, in JavaScript code, and so on. If the nature of the application is such that parts of the application cannot reside close enough to the user for the type of interaction required, then, of course, one should *not* simply have a slow version of (say) dragging an icon around, but instead change the overall interaction style to reflect the available resources.

Two of the factors that alleviate the effects of longer delays are predictability of the delay and progress indicators. Both factors give the user some sense of control or understanding over the process, especially if users have some indication of expected delays before initiating an action (Johnson' 1997). The many variable factors in networked systems make predicting delays very difficult, increasing the importance of giving users some sense of progress. The psychological effect of progress indicators is exploited (cynically) by those web browsers that have progress bars that effectively lie to the user, moving irrespective of any real activity (try unplugging a computer from the network and attempting to access a web page, some browsers will hit 70% on the progress bar before reporting a problem). Other network applications use recent network activity to predict remaining time for long operations (such as large file downloads). Other solutions include generating some sort of intermediate low quality or partial information while the full information is being generated or downloaded (e.g., progressive image formats or splash pages of Flash movies). If a progress bar only is available, research has shown that manipulations of the way it is presented (e.g., pulsating patterns on the bar) can significantly change the perceived wait time (Harrison, Yeo, and Hudson 2010).

For virtual reality using head-mounted displays, as well as hand–eye coordination tasks, we also have issues of the coordination between head movements and corresponding generated images. The timescales here are even tighter as the sensory paths are faster within our bodies, hence less tolerant of external delays. The brain receives various indications of movement: the position and changes of neck and related muscles, the balance sensors in the inner ear, and the visual feedback. Delays between the movement of the generated environment and head movement lead to dissonance between these different senses and have an effect rather like being at sea, with corresponding disorientation and nausea. Also any delays reduce the sense of immersion—being there within the virtual environment. Early studies of VR showed that users' sense of immersion was far better when they were given very responsive wire frame images than when they were given fully rendered images at a delayed and lower frame rate (Pausch 1991).

11.4.2.3 Coping Strategies

People are very adaptable. When faced with unacceptable delays (or other user interface problems) users develop ways to work around or ameliorate the problem—coping strategies. For example, web users may open multiple windows so that they can view one page while reading another (McManus 1997) and users of "telnet" for remote command line interfaces may type "test" characters to see whether the system has any outstanding input (Dix 1994).

Coping strategies may hide real problems, so it is important not to assume that just because users do not seem to be complaining or failing that everything is all right. However, we can also use the fact that users are bright and resourceful by building interface features that allow users to adopt coping strategies where it would be impossible or impractical to produce the interface response we would like. For example, where we expect delays we can ensure that continual interaction is not required (by perhaps amassing issues requiring user attention in a "batch" fashion), thus allowing users to more easily multitask. Unfortunately, this latter behavior is not frequently seen—the "myth" lives on and most networked programs still stop activity and await user interaction whenever problems are encountered.

11.4.2.4 Timeliness of Feedback/Feedthrough, Pace

Although feedback is one of the most heavily used terms in HCI, we may often ignore the complex levels of feedback when dealing with near instantaneous responses of GUI interfaces. In networked systems with potentially long delays, we need to unpack the concept. We have already discussed some of the critical timescales for feedback. For hand–eye coordination getting feedback of less than the 100-millisecond threshold is far more important than fidelity—quickly moving wire frames or simple representations are better than dragging an exact image with drop shadow.

For longer feedback cycles, such as pressing a button, we need to distinguish:

- Syntactic feedback—the system has recognized your action
- Intermediate feedback—the system is dealing with the request implied by your action (and if possible progress toward that request)
- Semantic feedback—the system has responded and the results obtained

The direct manipulation metaphor has led to identification and hence confusion between these levels, and many systems provide little in the way of syntactic or intermediate feedback relying solely on semantic feedback. In networked systems where the semantic feedback includes some sort of remote resource, it is crucial to introduce specific mechanisms to supply syntactic and intermediate feedback, otherwise the system may simply appear to have ignored the user's action (leading to repeated actions with potentially unforeseen consequences) or even frozen or crashed.

This also reminds us of a crucial design rule for slow systems: Wherever possible make actions idempotent—that is, invoking the same action twice, where possible, should have the same effect as a single action. This means that the "try again" response to a slow system does not lead to strange results.

For collaborative systems or those involving external or autonomous resources (remote-controlled objects, environmental sensors, software agents), we must also consider feedthrough. Feedback is experiencing the effect of one's own actions; feedthrough is the effect of one's own actions on other people and things and experiencing the effects of their actions themselves. For example, in an online chat system, you type a short message and press "send" and your message appears in your transcript—feedback—then sometime later it also appears in the transcript of the other chat participants—feedthrough.

Feedback is needed to enable us to work out whether the actions we have performed are appropriate, hence (typically) need to be much quicker than feedthrough responses. This is fortunate as feedthrough by its very nature usually requires network transmission and ensuing delays. The exception to the rule that feedthrough can afford to be slower is where the users are attempting to perform some close collaborative task (e.g., positioning some items using direct manipulation) or where there is a second-fast communication channel (e.g., on the telephone, user A says to user B "see the red box," but the relevant item has not appeared yet on B's screen).

Potentially more important to users of collaborative systems than bandwidth or even raw delays is pace—the rate at which it is possible to interact with a remote resource or person. This is partly determined by lower level timings, but is also heavily influenced by interface design. For example, you know that someone is sitting at his or her desk and you send him or her an urgent e-mail. The time that it takes to get a response will be hardly affected by the raw speeds between your machine and your colleague, and more determined by factors such as how often the e-mail client checks the server for new e-mail and whether it sounds an alert when new e-mail arrives, or simply waits there until your colleague chooses to check the inbox.

11.4.2.5 Race Conditions and Inconsistent Interface States

Alison and Brian are using an online chat program.

Alison writes "It's a beautiful day. Let's go out after work" and then begins to think about it.

Brian writes "I agree totally" and then has to leave to go to a meeting.

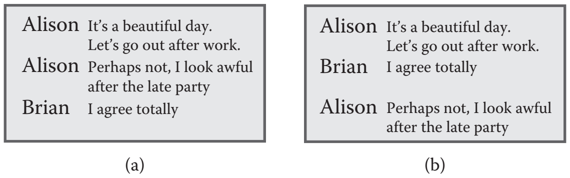
At almost the same time, Alison writes "Perhaps not, I look awful after the late party"

Unfortunately the messages are so close to simultaneous that both Alison and Brian's machines put their own contribution first so Alison sees the chat window as in Figure 11.9a, and Brian sees it as in Figure 11.9b. Brian thinks for a few moments, and then he writes "No you look lovely as ever," but unfortunately Alison never sees this as she takes one look at Brian's previous remark and shuts down the chat program.

This type of incident where two events happen so close together that their effects overlap is called a race condition. Race conditions may lead to inconsistent states for users as in this example, or may even lead to the software crashing. Although in principle race conditions are possible however fast the underlying network, the likelihood of races occurring gets greater as the network (and other) delays get longer.

Even some of the earliest studies in collaborative systems have shown the disorienting effects of users seeing different views of their shared information space, even when this is simply a matter of seeing different parts of the same space (Stefik et al. 1987).

Consistency becomes an even greater problem in mobile systems where wireless connections may be temporarily lost, or devices may be unplugged from fixed networks while on the move. During these periods of disconnection, it is easy for several people to be updating the same information leading to potential problems when their devices next become network connected.



Alison It's a beautiful day.
Let's go out after work.
Alison Perhaps not, I look awful
after the late party
Brian I agree totally

(a)

Alison It's a beautiful day.
Let's go out after work.
Brian I agree totally
Alison Perhaps not, I look awful
after the late party

(b)

FIGURE 11.9 Consistency breakdown. (a) Alison's chat window. (b) Brian's chat window.

In Section 11.6, we will discuss mechanisms and algorithms that can be used to maintain consistency even when delays are long and race conditions likely to occur.

11.4.2.6 Awareness

Returning to Alison and Brian, after Brian has typed his response, he may not know that Alison has not seen his second contribution.

Awareness of who is around and what they are doing is a major issue in CSCW (e.g., Dourish and Bellotti [1992] and McDaniel and Brinck [1997]). It has various forms:

- Being able to tell easily, when you want to know, what other people are doing
- Being made aware (through alerts, very salient visual cues, etc.) when significant events occur, for example, a new user arrives, someone makes a contribution
- Having a peripheral awareness of who is around and what they are up to

Awareness is not just about other people. In any circumstance where the environment may change, but not through your own direct action, you may need to know what the current state is and what is happening. This is not confined to networked applications, but applies to any hidden or invisible phenomena, for example, background indexing of your hard disk contents. In networked applications, anything distant is invisible unless it is made visible (audible) in the interface.

One of the earlier influential experiments to demonstrate the importance of peripheral awareness was ArKola (Gaver, Smith, and O'Shea 1991). This was a simulated bottling factory where two people worked together in maintaining the factory, supplying, maintaining the process, and so on. The participants could not see the entire factory at once, so relied on the sounds produced to be aware of its smooth running or if there are any problems. For example, the sound of breaking glass might suggest that the end of the production line has run out of crates, but if it immediately stopped, one would assume that the other participant had sorted out the problem.

The numerous forms of shared video and audio spaces are another example of this—several people, usually in distant offices establish these long-term, always-on audio, video, or audio–video links between their offices (Buxton and Moran 1990; Olson and Bly 1991). Sometimes these are used for direct communication, but most of the time they just give a peripheral awareness that the other person is there and the sort of activity they are doing. This can be used for functional purposes (e.g., knowing when the other person is interruptible), but also for social purposes—feeling part of a larger virtual office. Other systems have allowed larger numbers of, usually deliberately low quality and so less intrusive, webcam views of colleagues' offices and shared areas (Roussel 1999, 2001). The aim is the same, to build social cohesion, to allow at-a-glance reading of one another's situation, and to promote "accidental" encounters.

A form of awareness mechanism is now common on the web with buddy-lists that tell you when friends are online (ICQ 2001), microblogging such as Twitter feeds and "status" on social network sites such as Facebook. I still know of no examples of rich media experiments in a domestic environment, for example, a virtual kitchen shared with your elderly mother in Minnesota. However, various forms of domestic sharing through the Internet are becoming more common including the early Casablanca project at Interval (Hindus et al. 2001) using shared electronic sketch pads in the home and more recent projects (CASIDE 2005; Taylor et al. 2006). Some of these applications are fairly standard "computer" interfaces, some are soft surveillance, such as monitoring an elderly relative; however, there is also a stream of research looking at more intimate ways of sharing presence including pads that glow or warm when a far-away loved one touches them, or Jenny Tillotson's "Scent Whisper," a pair of Internet connected brooches which emit a pleasant scent when the loved one whispers on the other end (Tillotson 2005).

Trying to capture all this information within a computer display can be distracting, use up valuable screen space, and of course assumes that the computer is there and switched on. For this reason, several projects have looked at ambient interfaces, which in various ways make the physical environment reflect the virtual. These interfaces monitor various events in the electronic worlds and then change things in the physical environment: lights on the wall, moving strings hung from the ceiling, even a shaking potted plant (Lock, Allanson, and Phillips 2000). Again, this is not fundamentally limited to networked environments, but is of course not very useful when the relevant activity is close at hand anyway.

The other side of this is finding out what people are doing in order to signal this to others. For computer activity—are you logged on? have you been typing recently? what web page are you viewing?—this is in principle available, although the various layers of software may make it hard for an awareness service to discover. For noncomputer aspects, this is more problematic—are you in the room, busy, with other people—and may require a range of sensors in the environment: ultrasound, video, and so on, with corresponding privacy issues (see for example Bellotti [1993]). Monitoring of everyday objects is another way to achieve this, for example, one experiment used electronic coffee cups with sensors to tell when they were picked up and moved around (Gellersen, Beigl, and Krull 1999). As more and more devices become networked, it may be that we do not need special sensors, just use the combined information from those available, although the privacy issues remain.

In collaborative virtual reality environments, knowing that other people are around (as avatars) is as important as in a physical world, but harder due to limited senses (usually just vision and sound), limited field of view. Furthermore, there are computational costs in passing information such as audio or even detailed positional information around the network when there are tens, hundreds, or thousands of users. Various spatial models have been developed to analyze and implement the idea of proximity in virtual space (Benford et al. 1994; Rodden 1996; Sandor, Bogdan, and Bowers 1997; Dix et al. 2000, 2005; Dix 2009). These seek to formalize concepts of (1) where your focus of attention is within the virtual world and thus whether you require full quality audio and visual representation of others; (2) broader areas where you would expect some peripheral awareness where potentially degraded information can be used; and (3) those parts of the space for which you need no awareness information.

11.4.3 Media Issues

When describing the intrinsic network properties, issues for continuous media were mentioned several times. This is because, with the possible exception of close hand–eye coordination tasks, continuous media put some of the tightest requirements on the underlying networks.

11.4.3.1 Interactive Conversation and Action

Most demanding of all are audio–visual requirements of interactive conversation. Anyone who has had a transcontinental telephone conversation will have some feeling for the problems a delay of a second or two can cause. While actually speaking, the delays are less significant; however, turn-taking becomes very problematic. This is because the speaker in a conversation periodically (and subconsciously) leaves short (200–300 milliseconds) gaps in the flow of speech. These moments of silence act as entry points for the other participant who is expected to either acknowledge with a "go on" sound, such as "uhm," or perhaps a small nod of the head, or can use to break in with their own conversation. Entries at other points would be seen as butting in and rude, and lack of feedback responses can leave the speaker uncertain as to the listener's understanding. The 200–300 milliseconds is again almost certainly related to the time it takes for the listener's sensory system to get the relevant aural information to the brain, and for the brain to signal the relevant nod, acknowledgement, or start to speak. Clearly our conversational system is finely tuned to the expected intrinsic delays of face-to-face conversation.

When network delays are added, it is no longer possible to respond within the expected 200–300-millisecond window. The speaker therefore gets no responses at the appropriate points, and it is very hard for the listener to break into the flow of speech without appearing rude (by the time they hear the gap and speak, the speaker has already restarted). Some telephone systems are half-duplex; that is, they only allow conversation in one direction at a time, and this means that the various vocalizations ("uhu," "hmm," etc.) that give the speaker feedback will be lost entirely while the speaker is actually talking. It is not uncommon for the speaker to have to resort to saying "Are you there?" due to a loss of sense of presence.

These effects are similar, whether one is dealing with pure audio stream (as with the telephone), video streams (as with desktop conferencing), or distributed virtual environments. One VR project in the United Kingdom conducted all its meetings using a virtual environment in which the participants were represented by cuboid robot-like avatars (called blockies) (Greenhalgh 1997). The project ended with an online virtual party. As the music played the participants (and their avatars) danced. Although they were clearly enjoying, the video of the party showed an interesting phenomenon. Everyone danced alone! There are various reasons for this, for example, it was hard to determine the gender of a potential dancing partner. However, one relates directly to the network delays. Although everyone hears the same music, they all were hearing it at slightly different time; furthermore, the avatars for other people will be slightly delayed from their actual movements. Given popular music rhythms operate at several beats per second, even modest delays means that your partner appears to dance completely out of time! In more recent work, predictive algorithms have been used to create "ghost" figures showing the "best guess" location of people and things in collaborative virtual environments (Gutwin et al. 2004).

11.4.3.2 Reliability

As well as delays, we noted in Section 11.4.1 that network connections may not always be reliable—that is, information may be lost. Video and audio streams behave very differently in the presence of dropped information. Imagine you are watching a film on a long air flight. The break in the sound when the pilot makes an announcement is much more difficult than losing sight of the screen for a moment or two as the passenger in front stands up. At a smaller scale, a fraction of a second loss of a few frames of video just makes the movie seem a little jerky, and a smaller loss of even a few tens of milliseconds of audio signal would make an intrusive click or distortion. In general, reliability is more important for audio than video streams and where resources are limited, it is typically most important to reserve the QoS for the audio stream.

11.4.3.3 Sound and Vision

Why is it that audio is more sensitive than video? Vision works (largely) by looking at a single snapshot—try walking around the room with your eyes shut, but opening them for glances once or twice a second. Apart from the moment or two as your eyes refocus, you can cope remarkably well. Now turn a radio on with the sound turned very low and every second turn the sound up for a moment and back to silent—potentially an interesting remixing sound, but not at all meaningful. Sound more than vision is about change in time. Even the most basic sounds, pure tones, are measured in frequencies—how long between peaks and troughs of air pressure. For more complex sounds, the shape of the sound through time—how its volume and frequency mix changes—is critical. For musical instruments, it is hard to hear the difference between instruments if they are playing a continuous note, but instantly differentiable by their attack—how the note starts. (To get some idea of the complexity of sound, see the review in Mitsopoulos' thesis [Mitsopoulos 2000], and for an insight into the way different senses affect interaction, see my own AVI'96 article [Dix 1996].)

11.4.3.4 Compression

As we discussed in Section 11.4.1, it is possible to produce reliable network connections, but this introduces additional delays. Compression can also help by reducing the overall amount of audio–visual data that needs to be transmitted, but again may introduce additional delays. Furthermore, simple compression algorithms require reliable channels (both kinds of delays). Special algorithms can be designed to cope with dropped data, making sure that the most important parts of the signal are replicated or "spread out" so that dropped data leads to a loss in quality rather than interruption.

11.4.3.5 Jitter

As noted in Section 11.4.1, jitter is particularly problematic for continuous media. Small variations in delay can lead to jerky video play back, but is again even worse for audio streams. First of all, a longer than normal gap between successive bits of audio data would lead to a gap in the sound just like dropped data. And perhaps even more problematic, what do you do when subsequent data arrives closer together—play it faster? Changing the rate of playing audio data does not just make it jerky, but changes the frequency of sound rendering it meaningless.

(Aside: Actually, there are some quite clever things you can do by digitally speeding up sound but not changing its frequency. These are not useful for dealing with jitter, but can be used to quickly overview audio recordings, or catch up on missed audio streams [Arons 1997; Stifelman, Arons, and Schmandt 2001].)

For real-time audio streams, such as videoconferencing or Voice over IP, it is hard to do anything about this, and the best one can do is drop late data and do some processing of the audio stream to smooth out the clicks this would otherwise generate.

11.4.3.6 Broadcast and Prerecorded Media

Where media is prerecorded or being broadcast but where a few seconds delay are acceptable, it is possible to do far better. Recall that in several places we saw that better quality can be obtained if we are prepared to introduce additional delays.

If you have used streaming audio or video broadcasts, you will know that the quality is quite acceptable and does not have many of the problems described in Sections 11.4.3.1 through 11.4.3.5. This is partly because of efficient compression, meaning that video is compressed to a fraction of a percent of its raw bandwidth and so can fit down even a modem line. However, this would not solve the problems of jitter. To deal with this, the player at the receiving end buffers several seconds of audio–visual data before playing it back. The buffering irons out the jitter giving continuous quality.

Try it out for yourself—tune onto a radio channel and simultaneously listen to the same broadcast over the Internet with streamed audio. You will clearly hear up to a minute delay between the two.

Fast and free Internet connections have enabled the sharing and distribution of high-quality media for storing and playing later initially through illegitimate file-sharing including the pre-lawsuit Napster and many current peer–peer applications, but also increasingly through paid-for services such as the current Napster and Apple iTunes (Wikipedia contributors 2006). In addition to forcing existing media publishers to revisit their business models, the rise of web-distributed MP3 and Podcasting has enabled would-be artists and broadcasters to bypass the traditional distribution channels.

11.4.4 Public Perception: Ownership, Privacy, and Trust

One of the early barriers to consumer e-commerce was distrust of the transaction mechanisms—especially giving credit card details over the web. Arguments that web transactions are more secure than phone-based credit cards transactions or even using a credit card at your local restaurant (which gets both card number and signature) did little to alleviate this fear. This was never as major a barrier in the United States as it was, for example, in Europe, but across the world was a concern, slowing down the growth of e-shopping (or really e-buying, but that is another story [Dix 2001a]).

It certainly is the case that transactions through secure channels can be far more secure than physical transactions, where various documents can be stolen or copied en route and are in a format much more easy to exploit for fraud. However, knowing that a transaction is secure is more than the mechanisms involved, it is about human trust. Do I understand the mechanisms well enough, and the people involved well enough, to trust my money to it?

In fact, with a wider perspective, this distrust is very well founded. Encryption and authentication mechanisms can ensure that I am talking to a particular person or company and that no-one else can overhear. But how do I know to trust that person? Being distant means I have few of the normal means available to assess the trustworthiness of my virtual contact. In the real world, I may use the location and appearance of a shop to decide whether I believe it will give good service. For mail order goods, I may use the size, glossiness, and publication containing an advert to assess its expense and again use this to give a sense of quality of the organization. It is not that these physical indicators are foolproof, but that we are more familiar with them. In contrast, virtual space offers few intrinsic affordances. It is easy and quite cheap to produce a very professional web presence, but it may be little more than a facade. This is problematic in all kinds of electronic materials, but perhaps most obvious when money is involved.

As a designer, it is important not just to assure your users that your site is trustworthy, but also to make sure you do not create habits in your users that may be dangerous in the future. For example, e-mail "phishing" attacks often have links to sites at different domains from the sender address of the e-mail, but some e-mails from banks do exactly the same—they may be safe, but they create patterns of use that may mean you end up giving away your bank log-in information to another party. For e-mails on sites that need to be secure, it is often the best policy to simply ask users to go to your website and enter some form of quick access code rather than a clickable link.

Even if I trust the person at the other end, how do I know whether the network channel I am using is of a secure kind? Again the affordances of the physical world are clear: in a closed office versus in the open street versus in a bar frequented by staff of a rival firm. We will say different things depending on the perceived privacy of the location. In the electronic world, we rely on "https" at the beginning of a URL (how many ordinary consumers know what that means?), or an icon inserted by the e-mail program to say a message has been encrypted or signed. We need to trust not only the mechanisms themselves, but also the indicators that tell us what mechanisms are being used (Millett, Friedman, and Felton 2001; Fogg et al. 2001).

In nonfinancial transactions, issues of privacy are also critical. We have already seen several examples where privacy issues occur. As more devices become networked, especially through wireless links and our environment and even our own bodies become filled with interlinked sensors, issues about who can access information about you become more significant. This poses problems at a technical level—ensuring security among devices, at an interface level—being able to know and control what or who can see specific information, and at a perception level—believing in the privacy and security of the systems. There are also legal implications. For example, in the United Kingdom, in 2001, it was illegal for mobile telecoms operators to give location information to third parties (Sangha 2001).

The issue of perception is not just a minor point, but also perhaps the dominant one. Networks, and indeed computer systems in general, are by their nature hidden. We do not see the bits travelling down the wires or through the air from device to device, but have to trust the system at even the most basic level. As HCI specialists, we believe ourselves a little above the mundane software engineers who merely construct computer systems, as we take a wider view and understand that the interaction between human and electronic systems has additional emergent properties and that it is this complete socio-technical unit that achieves real goals. For networked systems, this view is still far too parochial.

Imagine if the personal e-mail of millions of people was being sucked into the databanks of a transnational computer company, and only being released when accessed through the multinational's own web interface. The public outcry! Imagine Hotmail, Yahoo! mail, and so on. How is it that although stored on distant computers, perhaps half the world away, millions of people feel that it is "their" mailbox and trust the privacy of web mail more than perhaps their organization's own mail system. This feeling of ownership of remote resources is more than the technology that protects the security of such systems; it is a cultural phenomenon and a marketing phenomenon. The web mail "product" is not just technology, or interface, but formed by every word that is written about the product in ads, press releases, and media interviews (Dix 2001a).

11.5 NETWORKS AS SUBJECTS: UNDERSTANDING AND MANAGING NETWORKS

When using a networked application, you do not really care what kind of network is being used, whether the data is sent over copper wires, fiber optic, microwave, or satellite. All you care about is that the two ends manage to communicate and the effects that any of the above have on the end-to-end network properties, such as bandwidth discussed in the previous section. However, there are times when the network's internals cannot be ignored.

Those involved in installing or managing networks need to understand the internal workings of the network in order to optimize performance and find faults. For ordinary users, when things go wrong in a networked application, they effectively become a network manager and so understanding something of the network can help them to deal with the problem. Even when things are working, having some awareness of the current state of the network may help one predict potential delays, avoid problems, and minimize costs. In some cases, this can even be used as a positive part of the interactive experience.

We will start this section by looking at some of the technical issues that are important in understanding networks. This parallels Section 11.4.2, but is focused on the internal properties of the network. We will then look at the interface issues for those managing networks and the ways in which interfaces can make users aware of critical network state. Finally, we will look briefly at the way models of networks can be used as a metaphor for some of the motor and cognitive behaviors of humans.

11.5.1 Network Models

11.5.1.1 Layers

Networking is dominated by the idea of layers—lower levels of the network offer standard interfaces to higher levels so that it is possible to change the details of the lower level without changing the higher levels. For example, imagine you are using your laptop to access the Internet while in a train (see Figure 11.10). The web browser establishes a TCP/IP connection to the web server, requests a web page, and then displays it. However, between your laptop and the web server, the message may have travelled through a Bluetooth link to your mobile phone, which then used a cell-based radio to send it to a mobile-phone station, and then through a microwave link to a larger base station onto a fiber-optic telephone network backbone and through various copper wires to your telecom service provider. Your service provider is then connected into another fiber-optic Internet backbone and eventually through more fiber-optic, copper, and microwave links to the web server. To complicate things even further, it may even be that the telephone and Internet backbones may share the same physical cabling at various points. Imagine if your poor laptop had to know about all of this.

In fact, even your PDA will know about at least five layers:

- Bluetooth—how the laptop communicates to the phone
- Modem—how the laptop uses the phone to link to your Internet service provider (ISP)
- IP—how your laptop communicates to the web server computer
- TCP—how data is passed as a reliable connection-based channel between the right program on your PDA and the web-server computer
- HTTP—how the browser communicates to the web server

Each of these hides most of the lower levels, so your browser needs to know nothing about IP, the modem, or the infrared connection while accessing the web page.

The nature of these layers differs both between different types of network, for example WAP, for sending data over mobile phones and devices, has five defined layers (Arehart et al. 2000), the ISO OSI reference model has seven layers (see Figure 11.11) (ISO/IEC7498 1994).

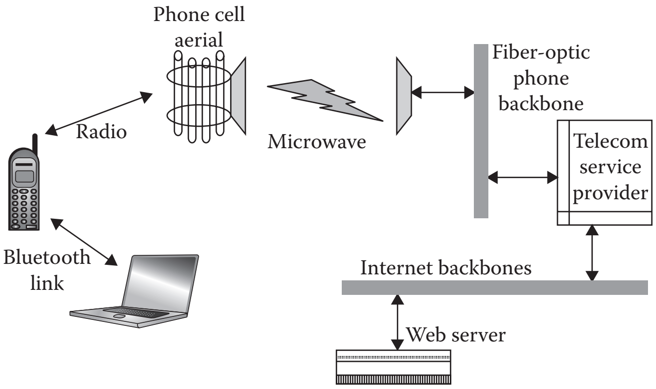


FIGURE 11.10 The long path from laptop to web.

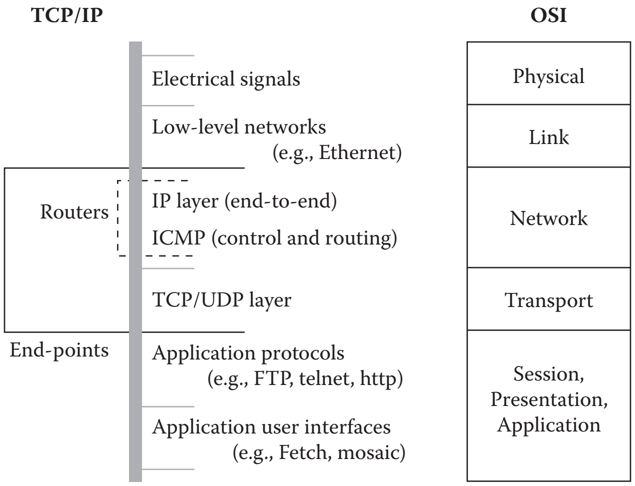


FIGURE 11.11 OSI seven layers and TCP/IP.

Mail program says	SMTP server replies
HELO mypc.mydomain.com	220 mail.server.net ESMTP
MAIL From:<myself@mydomain.com>	250 mail.server.net Hello mypc.mydomain.com[100.0.1.7], pleased to meet you
RCPT To: <a-friend@theirdomain.com>	250 <myself@mydomain.com>… Sender ok
DATA	250 <a-friend@theirdomain.com>… Recipient ok
	354 Enter mail, end with "." on a line by itself
..dotty 2nd line	first line of message
	last line
QUIT	250 KAA24082 Message accepted for delivery
	221 mail.server.net closing connection

FIGURE 11.12 Protocols to send e-mail through Simple Mail Transfer Protocol.

Header				Body
to address	from address	info	data length	data …..

FIGURE 11.13 Typical network packet format (simplified).

Ethernet header			Ethernet body			
			IP header			IP body
Ethernet to addr.	Ethernet from addr.	IP flag data len. etc.	IP to addr.	IP from addr.	Other header IP info	Internet data…..

FIGURE 11.14 Internet IP packet inside Ethernet packet.

11.5.1.2 Protocols

Systems at the same layer typically require some standard language to communicate. This is called a protocol. For higher levels, this may be quite readable. For example, to send an Internet e-mail message, your mail program connects to a simple mail transfer protocol (SMTP) server (a system that relays messages) using TCP/IP and has the exchange shown in Figure 11.12.

At lower levels, data is usually sent in small packets, which contain a small amount of data plus header information saying where the data is coming from, where it is going to, and other bookkeeping information such as sequence numbers, data length, and so on.

Even telephone conversations, except those using predigital exchanges, are sent by chopping up your speech into short segments at your local telephone exchange, sending each segment as a packet and then reassembling the packets back into a continuous stream at the other end (Stevens 1998, 1999).

11.5.1.3 Internetworking and Tunneling

This layering does not just operate within a particular network standard, but between different kinds of networks too. The Internet is an example of an internet (notice little "i") that is a network which links together different kinds of low-level network. For example, many PCs are connected to the Internet through an Ethernet cable. Ethernet sends its own data in packets like those of Figure 11.13. The IP also has packets of a form like Figure 11.13. When you establish an Internet connection through Ethernet, the IP packets are placed in the data portion of the Ethernet packet, so you get something a bit like Figure 11.14.

This placing of one kind of network packet inside the data portion of another kind of network is also used in VPNs in a process called tunneling. These are used to allow a secure network to be implemented using a public network like the Internet. Imagine a company has just two offices, one in Australia and the other in Canada. When a computer in the Sydney office sends data to a computer in the Toronto office, the network packet is encrypted, put in the data portion of an Internet packet, and sent through the Internet to a special computer in the Toronto office. When it gets there, the computer at the Toronto office detects it is VPN data, extracts the encrypted data packet, decrypts it and puts it onto its own local network where the target computer picks it up. As far as both ends are concerned, it appears as if both offices are on the same LAN and any data on the Internet is fully encrypted and secure.

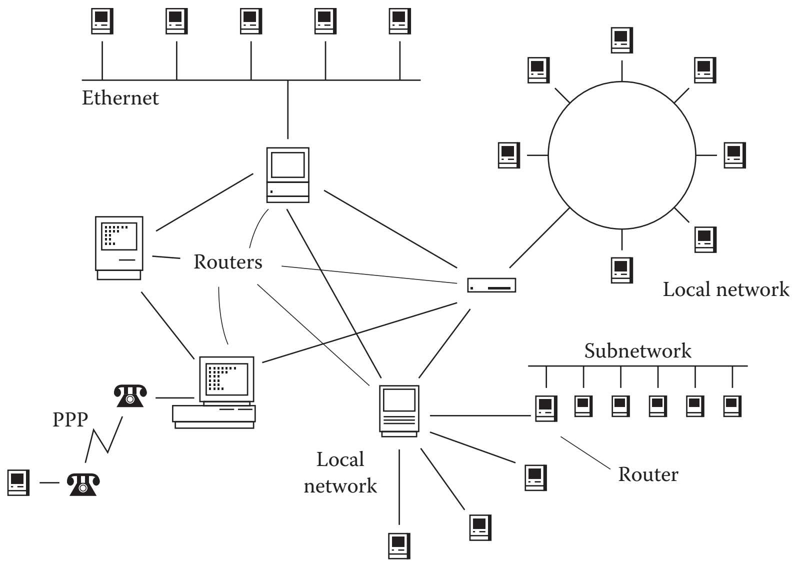


FIGURE 11.15 Routers send messages in the right direction through complex networks.

11.5.1.4 Routing

If two computers are on the same piece of physical network, each can simply listen out for packets that are destined for them, so sending messages between them is easy. If however messages need to be sent between distant machines, for example, if you are dialed into an ISP in the United Kingdom and are accessing a web server in the United States, the message cannot simply be broadcast to every machine on the Internet (Figure 11.15). Instead at each stage, it needs to be passed in the right direction between different parts of the network. Routers perform this task. They look at the address of each packet and decide where to pass it to. If it is a local machine, this might be to simply put it onto the relevant local network, but if not it may need to pass it on to another intermediate machine.

Routers may be standalone boxes in network centers, or may be a normal computer. Often a file server acts as a router between a LAN and the global network.

Besides routers, networks are also linked by hubs and switches, which make several different pieces of physical network behave as if they were one local network, and gateways that link different kinds of network. The details of these are not important, but they add more to the sheer complexity of even small networks.

11.5.1.5 Addresses

In order to send messages on the Internet or any other network, you need to have the address of where they are to go (or at least your computer needs it). In a phone network, this is the telephone number, and on the Internet, it is an IP number. The IP number is a 32-bit number, normally represented as a group of four numbers between 0 and 255 (e.g., 212.35.74.132), which you will have probably seen at some stage when using a web browser or other Internet tool. It is these IP numbers that are used by routers to send Internet data to the right place.

> The 32-bit IP number space allows for 4 billion addresses. This sounds like quite a lot; however, these have been running out because of the explosive growth in the number of Internet devices and "wasted" IP numbers due to the way ranges of numbers get allocated to subnetworks. The new version of TCP/IP, IPv6, which is being deployed, has 128-bit addresses, which requires 16 numbers (IPng 2001). This allows sufficient unique IP addresses for every phone, PDA, Internet-enabled domestic appliance, or even electronic paper clip.

However, with any network, there is a problem of how you get to know the address. With phone numbers, you simply look up the person's name in a telephone directory or by phoning the operator. Similarly, most networks have a naming scheme and some way to translate these into addresses. In the case of the Internet, domain names (e.g., acm.org, www.hcibook.com. magisoft.co.uk) are the naming system. There are so many of these and they are changed relatively rapidly, so there is no equivalent of a telephone directory, but instead special computers called domain name servers (DNS) act as the equivalent of telephone directory enquiries operators. Every time an application needs to access a network resource using a domain name (e.g., to look up a URL), the computer has to ask a DNS what IP address corresponds to that domain name. Only then can it use the IP address to contact the target computer.

The DNS system is an example of a "white pages" system. You have an exact name and want to find the address for that name. In addition, there are so-called "yellow pages" systems where you request, for example, a color postscript printer and are told of addresses of systems supplying the service. Sometimes, this may be mediated by brokers who may attempt to find the closest matching resource (e.g., a non-PostScript color printer) or even perform translations (e.g., the Java JINI framework [Edwards and Rodden 2001]).

This latter form of resource discovery system is most important in mobile systems and ubiquitous computing where we are particularly interested in establishing connections with other geographically close devices.

A final piece in the puzzle is how one gets to know the address of the name server, directory service, or brokering service. In some types of network, this may be managed by sending broadcast requests to a network "is there a name server out there." In the case of the Internet, this is normally explicitly set for each machine as part of its network settings.

11.5.1.6 All Together…

If you are in your web browser and you try to access the URL http://www.meandeviation.com/qbb/, the following stages happen:

1. Send IP-level request to DNS asking for www.meandeviation.com.
2. Wait for reply.
3. DNS sends reply 64.39.13.108.
4. Establish TCP level connection with 64.39.13.108.
5. Send HTTP request ("GET/qbb/HTTP/1.1").
6. Web server sends the page back in reply.
7. Close TCP connection.

Most of these stages are themselves simplified, all will involve layering on top of lower level networks and most stages involve several substages (e.g., establishing a TCP-level connection requires several IP-level messages).

The basic message is that network internals are multilevel, multistage, and pretty complicated.

11.5.1.7 Decentralizing: Peer–Peer and Ad Hoc Networks

Traditional Internet and web applications tended to work through client–server paradigms where a user's client application accesses a well-known such as a web server. However, there has been a growth of more decentralized models at both higher and lower protocol levels (Androutsellis-Theotokis and Spinellis 2004).

At an application level, peer–peer file sharing works by having clients running on users' own PCs talk directly to one another broadcasting requests for particular files by name or type to all connected clients. These then forward the requests to their connected nodes until some maximum hop count is hit. Often semi-centralized services are used to establish initial connections, but thereafter everything is done at the "edges" of the network. These protocols and applications have some technical strength, but their origins are rooted in legal disputes on the distribution of copyright material. Recent research has also shown that the anonymity of peer–peer file sharing also allows "deviant" subgroups sharing illicit material (Hughes et al. 2006).

At a lower level, ad hoc networks build more structured higher level networks for computers that have point-to-point connections, often through wireless connectivity. This can be used to set up a network in a meeting, or at a larger level could even allow everyone in a pop festival to have Internet access through a small number of wireless hot spots. In an ad hoc network, the low-level software works out what machines are connected where and routes messages from end to end even if the two end point computers do not have direct access. The difficult thing in an ad hoc network is not just setting up this routing information, but the dynamics: dealing with the fact that machines constantly move around, perhaps going out of range of one another, may be turned off, and join the network late.

11.5.2 Network Management

Those most obviously exposed to this complexity are the engineers managing large national and international networks, both data networks, such as the main Internet backbones, and telecoms networks. The technical issues outlined in Section 11.5.1 are compounded by the fact that the different levels of network hardware and network management software are typically supplied by different manufacturers. Furthermore, parts of the network may be owned and managed by a third party and shared with other networks. For example, a trans-Atlantic fiber optic cable may carry telecoms and data traffic from many different carriers.

When a fault occurs in such a network, it is hard to know whether it is a software fault or a hardware fault, where it is happening and who is responsible for it. If you send engineers out to the wrong location, it will cost both their time and also increase the time the service is unavailable. Typically, the penalties for inoperative or reduced quality services are high—you need to get it right fast.

This is a specialized and complex area, but clearly of increasing importance. It poses many fascinating UI challenges: How to visualize complex multilayered structures, and how to help operators trace faults in these. Although I know that it is a topic being addressed by individual telecoms companies, published material in the HCI literature is minimal. The exception is visualization of networks, both physical and logical, which is quite extensive (Dodge and Kitchin's Atlas of Cyberspace and associated website [Dodge and Kitchin 2001] is the comprehensive text in this area).

Ordinary systems administrators in organizations face similar problems albeit on a smaller scale. A near universal experience of misconfigured e-mail systems, continual network failures, and performance problems certainly suggests this is an area ripe for effective interface solutions, but again there is very little in the current HCI literature.

Finally, it appears that everyone is now a network administrator; even a first-time home PC user must manage modem settings, name server addresses, SMTP and POP servers, and more. It is interesting that for most such users, the interface they use is identical to that supplied for full systems administrators. Arguably this may ease the path for those who graduate from single machines to administering an office or organization, perhaps less than 5% of users. Unfortunately, it makes life intolerable for the other 95%! The only thing that makes this possible at all is that the "welcome" disks from many ISPs and the Wizards shipping with the OS offer step-by-step instructions or may automatically configure the system. These complications are compounded if the user wishes to allow access through more than one ISP, or connect into a fixed network. As many home users now have several PCs and other devices that need to be networked, this is not a minor issue.

In the first edition of this handbook, I wrote "If we look at the current state of the two most popular PC systems, Microsoft Windows and Mac OS, the picture is not rosy." Sadly things have not improved dramatically—you only have to watch a room full of computer scientists at a meeting trying to sort out wireless connections.

Some things have got better, for example, on MacOSX pretty much all settings are in one place under "Network Settings." Changing locations with a laptop is simply a matter of selecting "Location …" from the main menu. Similarly, the iPhone connects and reconnects largely seamlessly. However, ease of user brings its own problems as it can be very difficult to diagnose problems when they occur; the interfaces hide irrelevant detail to make life easier, but that "irrelevant" detail often becomes crucial in order to fix problems. Where information is available, it is buried in dialog boxes full of technical networking language.

These problems serve partly to emphasize the intrinsic complexity of networking—yes it does involve multiple logically distinct settings, many of which relate to low-level details. However, it also exposes the apparent view that those involved in network administration are experts who understand the meaning of various internal networking terms. This is not the case even for most office networks and certainly not at home.

And this is just initially setting up the system to use. For the home user, debugging faults has many of the same problems as large networks. You try to visit a website and get an error box … Is the website down? Are there problems with the wireless LAN, the broadband modem, the phone line, the ISP's hardware? Are all your configurations settings right? Has a thunderstorm 3000 miles away knocked out a vital network connection? Trying to understand a multilayered, nonlocalized and, when things work, largely hidden system is intrinsically difficult and where diagnostic tools for this are provided, they assume an even greater degree of expertise.

One example on the iPhone is that, when available, it uses Wi-Fi to connect to Internet service in preference to mobile broadband. This is done in order to speed up access and reduce load on the mobile network, both improving user experience and reducing the telecom operator's costs. When this works, it is fine, but sometimes applications such as the Facebook app simply fail "No Internet Connection" even though the phone appears to have full mobile broadband connectivity. One reason for this is when there is a Wi-Fi network available, which appears to be open, but actually has a web form requesting a log-in (and/or payment). If you are attempting to browse the web, this is immediately obvious and so you can either log-in/pay or go to the phone settings and turn off the Wi-Fi. However, when apps attempt to access the network, they are blocked and simply fail.

The promise of devices that connect up to one another within our homes and about our bodies is going to throw up many of the same problems. Some old ones may ease as explicit configuration becomes automated by self-discovery between components, but this adds further to the hiddenness and thus difficulty in managing faults, security, and so on. You can imagine the scenario—the sound on my portable DVD stops working and produces a continuous noise—why? There are no cables to check of course (wireless networking), but hours of checking and randomly turning devices on and off narrows the problem down to a fault in the washing machine which is sending continuous "I finished the clothes" alerts to all devices in the vicinity.

11.5.3 Network Awareness

One of the problems noted in Section 11.5.2 is the concealment of networks. This causes problems when things go wrong as one does not have an appropriate model of what is going on, but also sometimes even when things are working fine.

We discussed in Section 11.4.1 some of the network properties that may affect usability: bandwidth, delay, jitter, and so on. These are all affected to some extent by other network load, the quality of current network connections, and so on. So predicting performance (and knowing whether or not to panic if things appear to go slow) needs some awareness of the current state of the network.

PCs using wireless networks usually offer some indication of signal strength (if one knows where to look and what it means) although this is less common for line quality for modems. As wireless devices and sensors become smaller, they will not have suitable displays for this and explicitly making users aware of the low-level signal strength of an intelligent paper clip may not be appropriate. However, as interface designers, we do need to think how users will be able to cope and problem solve in such networks.

Only more sophisticated network management software allows one to probe the current load on a network. This does matter. Consider a small home network with several PCs connected through a single modem. If one person starts a large download, they will be aware that this will affect the performance of the rest of their web browsing. Other members of the household will just experience a slowing down of everything and not understand why. If the cause is a file-sharing utility such as Gnutella, even the person whose computer is running it may not realize its network impact. In experiments at MIT, the level of network traffic has been used to "jiggle" a string hanging from the ceiling so that heavy traffic leads to a lot of movement (Wisneski et al. 1998). The movements are not intrusive so give a general background awareness of network activity. Although supplying a ceiling-mounted string may not be the ideal solution for every home, other more prosaic interface features are possible.

Cost awareness is also very important. In the United Kingdom, the first-generation GSM mobile data services are charged by connection time. So knowing how long you have been connected and how long things are likely to take becomes critical. If these charges differ at peak hours of the day, calculating whether to read your e-mail now, or do a quick check and download the big attachments later can become a complex decision. The move to data-volume-based charging means the user needs to estimate the volume of data that is likely to be involved in initiating an action versus the value of that data—do you want to click on that link if the page it links to will include large graphics, perhaps an applet or two? Terrestrial broadband networks have largely shifted to fixed monthly fees, and this has dramatically changed the way people perceive the Internet from smash-and-grab interactions to surf-and-play. However, when roaming abroad, the charging regimes are still often usage-based and so this very free use of mobile Internet is still not ubiquitous.

11.5.3.1 Network Confusion

If the preceding does not sound confusing enough, the multilayered nature of networked applications means that it is hard to predict the possible patterns of interference between things implemented at different levels or even at the same level. Again this is often most obvious when things go wrong, but also because unforeseen interactions may mean that two features that work perfectly well in isolation may fail when used together.

This problem, feature interaction, has been studied particularly in standard telecoms (although certainly not confined to it). Let us look at an example of feature interaction. Telephone systems universally apply the principle that the caller, who has control over whether and when the call is made, is the person who pays. In the exceptions (free-phone numbers, reverse charges) special efforts are made to ensure that subscribers understand the costs involved. Some telephone systems also have a feature whereby a caller who encounters a busy line can request a call-back when the line becomes free. Unfortunately at least one company implemented this call-back feature so that the charging system saw the call-back as originating from the person who had originally been called. Each feature seemed to be clear on its own, but together meant you could be charged for calls you did not want to make.

With N features, there are $N(N-1)/2$ possible pairs of interactions to consider, $N(N-1)(N-2)/6$ triples, and so on. This is a well-recognized (but not solved) problem with considerable efforts being made using, for example, formal analysis of interactions to automatically detect potential problems. It is worth noting that this is not simply a technical issue; the charging example shows that it is not just who pays that matters, but the perceptions of who pays. This particular interaction would have been less of a problem if the interface of the phone system had, for example, said (in generated speech) "you have had a call from XXX press call back, you will be charged for this call." Although this issue has been most widely recorded in telecoms (Calder et al. 2003), these sorts of problems are likely to be found increasingly in related areas such as ubiquitous computing and resource discovery.

11.5.3.2 Exploiting the Limitations: Seamfulness and Virtual Locality

Some companies cynically exploit this user confusion over network charging, and in the United Kingdom, there have been some high-profile news stories about teenagers running up thousands of pounds of debt after innocently signing up to ring-tone delivery services.

Happily there are also more positive uses of the limitations of networks. Given suitable awareness mechanisms for network strength and connectivity, people are very resourceful in exploiting these. You will have experienced the way mobile phone users get to know the sweet spots for their networks when in areas of poor coverage, learning to get out of doors, away from big buildings or up hills. This can get sophisticated. One mixed-reality experiment, part of the Equator project, consisted of an outside game where real players running in the street were pitted against virtual characters manipulated remotely. The "real" players learned to hide in the global positioning system shadow of buildings so that their location could not be detected by the virtual participants (Benford et al. 2003). Similar effects have been seen in Wi-Fi-based games where participants get to know the regions of good and bad coverage and may use these to seek out or avoid remote interactions.

These are ways in which users do more than cope but actively exploit the limitations of networks and sensing. The Pirates game did this with RF technology; RF beacons represent islands in an ocean the locality of the RF signals corresponding to the limited land area of the island (Björk et al. 2001). This notion of exploiting network limitations has led to the idea of seamful games—deliberately designing so that variations in Wi-Fi coverage and connectivity are part of the gameplay (Chalmers et al. 2005).

Even more strange is a recent game "hitchers." In this, you pick up and drop off hitchers with your phone, just like you might pick up a hitch-hiker on the road. You can only pick up a hitcher in the same mobile cell as the last person dropped the hitcher off. The hitchers are actually stored and managed centrally, but the effect is as if the hitchers were only accessible in a small region. Here, even though there is no real limitation of access, a limitation is constructed to make a more interesting game—virtual locality. You can imagine virtual locality being used in synchronous applications, for example, allowing a phone user to broadcast to people in their vicinity apparently as if it were a limited range transmission, whereas in reality, it is centrally managed, based on global positioning system or phone cell location.

11.5.4 Network Within

So far the story is pretty bleak from a user interface viewpoint—a complex problem, of rapidly growing importance, with relatively little published work in many areas. One good thing as a HCI practitioner about understanding the complexity of networks is that they help us understand better the workings of the human cognitive and motor system.

For at least five decades, computational models have been used to inspire cognitive. Also, of course, cognitive and neurological models have been used to inspire computational models in artificial intelligence and neural networks. However, our bodies are not like a single computer, but in various ways more like a networked system.

First, the body is like a networked system because several things can happen at once. The interacting cognitive subsystems model from APU Cambridge (Barnard and May 1995) considers looking at various parts of the cognitive system, the conversions between representations between these parts and the conflicts that arise if the same part is used to perform different tasks simultaneously. Similarly, the very successful PERT-style goals, operators, methods, and selection rules analysis used on the NYNEX telephone operators interface used the fact that the operator could be doing several things simultaneously with no interfering parts of their bodies and brains (John 1990; Gray, John, and Atwood 1992).

We are also like a networked system in that signals take an appreciable time to get from our senses to our brains and from our brains to our muscles. The famous homunculus from Card, Moran, and Newell's (1983) Model Human Processor makes this very clear with timings attached to various paths and types of mental processing. In fact, the sorts of delays within our bodies (from 50–200 milliseconds on different paths) are very similar to those found on international networks.

In industrial control, one distinguishes between open-loop and closed-loop control systems (see Figure 11.16). Open-loop control is where you give the machine an instruction and assume it does it correctly (like a treasure map—"ten steps North, turn left, three steps forward and dig"). This assumes the machine is well calibrated and predictable. In contrast, closed-loop control uses sensors to constantly feedback and modifies future actions based on the outcomes of previous ones (e.g., "follow the yellow brick road until you come to the emerald city"). Closed-loop control systems tend to be far more robust, especially in uncontrolled environments, like the real world.

Not surprisingly our bodies are full of closed-loop control systems (e.g., the level of carbon dioxide in your lungs triggers the breathing reflex). However, closed-loop control can become difficult if there are delays—you have not received feedback from the previous action when starting the next one. Delays either mean one has to slow down the task or use some level of prediction to determine what to do next, based on feedback of actions before the last one. This breakdown of closed-loop control in the face of (especially unexpected) delays is one of the reasons hand–eye coordination tasks, such as mouse movement, breakdown if delays exceed a couple of hundred milliseconds (Section 11.4.2). The feedback loops in our bodies for these tasks assume normal delays of around 200 milliseconds and are robust to variations around this figure, but adding delays beyond this start to cause breakdown.

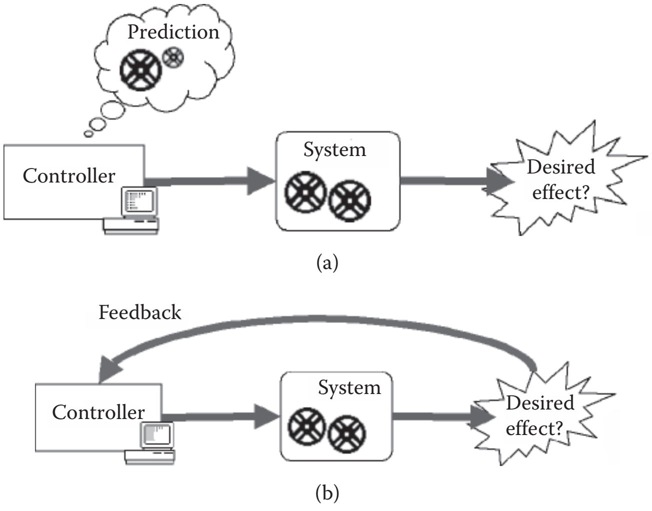


FIGURE 11.16 (a) Open-loop control. (b) Closed-loop control.

The delays inside our bodies cause other problems too. The path from our visual cortex into our brain is far faster (by 100 milliseconds or so) than that from our touch and muscle tension sensors around our bodies. If we were designing a computer system to use this information, we might consider having a short 100-millisecond tape loop, so that we could store the video input until we had the appropriate information from all senses. However, the sheer volume of visual information means that our brains do not attempt to do this. Instead, there is a part of our brains that predicts where it "thinks" our bodies are and what it is feeling based on previous nerve feedback and what it knows the muscles have been asked to do. The same bit of the brain then monitors what actually did happen (when the nerve signals have made their way up the spinal column to the brain) and gives either an uncomfortable or shocked sensation when a mismatch occurs. For example, you go to pick something up, but because of poor light or a strange shaped object, you touch it earlier or later than you would expect. Tickling is also connected with this lack of ability to predict the sensations (this is why it is difficult to tickle yourself).

Race conditions also occur within this networked system of our bodies—for example, getting letter inversions while typing where signals to the two hands get processed in the wrong order. Dix and Brewster (1994) also used race conditions to understand what goes wrong in certain kinds of mis-hits of onscreen buttons. In certain circumstances, two almost simultaneous "commands" from our brain to our hand to release the mouse button and to our arm to move to a new mouse location can get out of order meaning the mouse moves out of the target before it is released. This analysis allowed us to design an experiment that forced this very infrequent error to occur much more frequently and therefore make it easier to assess potential solutions.

11.6 NETWORKS AS PLATFORMS: ALGORITHMS AND ARCHITECTURES FOR DISTRIBUTED INTERFACES

User interfaces are hard enough to construct on a single machine, concurrent access by users on networked machines is a nightmare!

Happily, appropriate algorithms, architectures, toolkits, and frameworks can help … a bit.

11.6.1 Accessing Shared Objects

We saw in Section 11.4.2 how race conditions within networked systems can lead to inconsistencies within the user interface and within the underlying data structures. Fortunately, there are a range of techniques for dealing with this.

11.6.1.1 Locking

The standard technique, used in databases and file systems, for dealing with multiple accesses to the same data object is locking. When a user's application wants to update a particular database record, it asks the database manager for a lock on the record. It can then get a copy of the record and send back the update knowing that nothing else can access it in the meantime. Users are typically unaware that locking is being performed; the act of opening a file or opening an edit screen for a database record establishes the lock and later, when the file or the edit form is completed and closed, the lock is released.

Although this is acceptable for more structured domains, there are problems in more dynamic domains such as shared editing. Locking a file when one user is editing it is no good as we want several people to edit the same file at the same time. In these cases, more lightweight forms of locking can be used at finer granularities: at paragraph, sentence, or even per character level. For example, the act of clicking over a paragraph to set a text entry position may implicitly request a paragraph lock, which is released when you go on to edit another paragraph. However, implicit and informal locks, because they are not apparent, can lead to new problems. For example, a user may click on a paragraph, do some changes, but before moving on to another part of the document gets interrupted. No one else can then edit the paragraph. To avoid this, the more informal locks are often time limited or can be forcibly broken by the server if another user requests a lock on the same object.

11.6.1.2 Replication

In collaboration systems such as Lotus Notes/Domino or source code systems such as Subversion (SVN), users do not lock central copies of data, but instead each user (or possibly each site) has their own replica of the complete Notes database/SVN tree. Periodically, these replicas are synchronized with central copies or with each other. Updates can happen anywhere by anyone with a replica. Conflicts may, of course, arise if two people edit the same note between synchronizations. Instead of preventing such conflicts, the system (and software written using it) accepts that such conflicts will occur. When the replicas synchronize, conflicts are detected and various (configurable) actions may occur: flagging the conflicts to users adding conflicting copies as versions, and so on.

This view of replicate and worry later is essential in many mobile applications as attempts to lock a file while disconnected would first of all require waiting until a network connection could be made, and, worse, if the network connection is lost while the lock is still in operation could lead to files being locked for very long periods. Other examples of replication in research environments have included CODA at Carnegie Mellon University, which allows replication of a standard UNIX file system (Kistler and Satyanarayanan 1992) and Liveware, a contact information system that replicates and synchronizes when people meet in a manner modeled after the spread of computer viruses (Witten et al. 1991).

Commercial products offering data synchronization on PDAs and mobile phones has been common for many years, for example, HotSync was a key element of Palm OS since the mid-1990s. Now not only is this found on mobile devices to enable them to synchronize with desktop computers, but also on desktop applications synchronizing with large central servers or cloud data services to enable "anywhere" access to data; for example, FireFox Weave allows you to access shared bookmarks and other FireFox data from browsers on different machines (Mozilla Labs 2010).

Despite so many years of research and use, data synchronization still has significant problems. One example is in MacOSX, which has a very well-developed mechanism for third-party applications to synchronize between desktop machines and the iPhone, and also with the MobileMe cloud service (Apple 2010). However, if one attempts a three-way synchronization by enabling synchronization both directly between iPhone–desktop and through MobileMe, the algorithms used fail dramatically duplicating address book items and calendar events.

It is also interesting that commercial synchronization solutions rarely deal actively with conflicting updates, usually simply taking the last update as canonical. This may be because of technical reasons, if applications do not store fine-grained information about update and synchronization times, but also probably because it is difficult to design user interfaces to offer users the ability to decide between conflicts.

11.6.1.3 Optimistic Concurrency for Synchronous Editing

The "do it now and see if there are conflicts later" approach is also called optimistic concurrency, especially in more synchronous settings. For example, in a shared editing system, the likelihood of two users editing the same sentence at the same time is very low. An optimistic algorithm does not bother to lock or otherwise check things when the users start to edit in an area, but in the midst or at the end of their edits checks to see if there are any conflicts and attempts to fix them.

There are three main types of data that may be shared:

Orthogonal data—where the data consists of attributes of individual objects/records that can all be independently updated

Sequential data—particularly text, but any form of list where the order is not determined by an attribute property

Imagine two users, Adonis and Beatrice.

They are working using a shared editor and their current document reads as follows:

```
Adonis is ⊤ and Beatrice is ⊤ .
          A                 B
```

The sentence is partial and both users are about to type in their prime personal characteristic in order to complete it.

Adonis' insertion point is denoted by the boxed $\boxed{A}$ and Beatrice's insertion point is the boxed $\boxed{B}$.

Beatrice types first yielding the following:

```
Adonis is⊤and Beatrice is beautiful⊤.
         A                         B
```

Adonis then types "adorable", but unfortunately the implementor of the group editor was not very expert and the resulting display was:

```
Adonis is adorable⊤and Beatrice is ⊤ beautiful.
                  A                B
```

Beatrice's insertion point followed the thirty-sixth character before Adonis' insertion, and followed the thirty-sixth character after.

Reasonable but wrong! The actual text should clearly read as follows:

```
Adonis is adorable⊤and Beatrice is beautiful⊤.
                  A                         B
```

This correct behaviour is called a dynamic pointer as opposed to the static pointer "character position 36".

FIGURE 11.17 Dynamic pointers from Dix (1995).

Complex structural data—such as directory trees, taxonomic categories, etc.

In terms of complexity for shared data, these are in increasing difficulty.

Orthogonal data, although by no means trivial, is the simplest case. There is quite a literature on shared graphical editors, which all have this model—independent shapes and objects with independent attributes such as color, size, and position. When merging updates from two users, all one has to do is look at each attribute in turn and see whether it has been changed by only one user, in which case the updated value is used, or if it has been changed by both, in which case either the last update is used or the conflict is flagged.

Structured data is most complicated—What do you do if someone has created a new file in directory D, but at the same time someone else has deleted the directory? I know of no optimistic algorithms for dealing effectively with this in the CSCW literature. CODA deals with directory structures (normal UNIX file system) but takes a very simple view of this, as it only flags inconsistencies and does not attempt to fix them.

Algorithms for shared text editing sit somewhere between the two and have two slightly different problems, both relating to race conditions when two or more users are updating the same text:

Dynamic pointers—If user A is updating an area of text in front of user B, then the text user B is editing will effectively move in the document.

Deep conflict—What happens if user A's and user B's cursors are at the very same location and they perform insertions/deletions?

Figure 11.17 shows an example of the first of these problems. The deeper conflict occurs when both cursors are at the same position; say after the "Y" in "XYZ." Adonis types "A" and at the same time Beatrice types "B" should we have "XYABZ" or "XYBAZ" or perhaps even loose one or other character? Or if Adonis types "A" and Beatrice presses "delete" should we have "XYZ" or "XAZ"?

A number of algorithms exist for dealing with this (Sun and Ellis 1998; Mauve 2000; Vidot et al. 2000) including a retrofit to Microsoft Word called CoWord (Xia et al. 2004). Most of these stem from the dOPT algorithm used in the Grove editor (Ellis and Gibbs 1989). These algorithms work by having various operation transformations that allow you to reorder operations. For example, if we have two insertions (labeled a and b) performed at the same time at different locations:

a. Insert texta at location n
b. Insert textb at location m

but decide to give insert a preference, then we have to transform b to b' as follows:

b'. if (m < n) insert textb at location m (i)
if (m = n) insert textb at location m (ii)
if (m > n) insert textb at location m + length(texta) (iii)

Case (i) says that if the location of insert "b" is before insert "a" you do not have to worry. Case (iii) says that if it is after insert "a" you have to shift your insert along accordingly. Case (ii) is the difficult one where a conflict occurs and has to be dealt with carefully to ensure that the algorithm generates the same results no matter where it is. The version above would mean that B's cursor gets left behind by A's edit. The alternative would be to make case (ii) the same as case (iii), which would mean B's cursor would be pushed ahead of A's.

In early work in this area, I proposed regarding dynamic pointers as first class objects and using these in all representations of actions (Dix 1991, 1995). This means that rules like case (i) and case (iii) happen "for free," but the deep conflict case still needs to be dealt with specially.

Although operation transform methods have been in the research literature for many years, they have mainly been used in research systems. However, Google Wave uses operation transformation to enable fine-grained remote collaboration (Wang and Mah 2010), so this technology is eventually entering the mainstream.

11.6.1.4 Groupware Undo

The reason that undo is complicated in groupware is similar to the problems of race conditions in optimistic concurrency.

In the case of optimistic concurrency, user A has performed action a, and user B has performed action b, both on the same initial state. The problem is to transform user B's action into one b' that can be applied to the state after action a yet still mean the "same things" as the original action b (see Figure 11.18 left).

In the case of groupware undo, we may have the situation where user A has performed action a, followed by user B performing action b, and then user A decides to undo action a. How do we transform action b so that the transformed b' means the same before action a as b did after (see Figure 11.18 right).

Similar, but slightly different transformation rules can be produced for the case of undo and also dynamic pointers can be used for most cases.

As with optimistic concurrency, there is slightly more work on group undo in shared graphical editors where the orthogonal data makes conflicts easier (Berlage and Spenke 1992).

11.6.1.5 Real Solutions?

Although these various algorithms can ensure there is no internal inconsistency and that all participants see the same thing, they do not necessarily solve all problems. Look again at the case of Alison and Brian's chat in Section 11.4.2. Certainly in the case of group undo, when Abowd and Dix (1992) published the first article on the topic, we proposed various solutions, but also recommended that besides an explicit undo button, systems ought to provide sufficient history to allow users to recreate what they want to without using the undo button. This is not because it is impossible to find a reasonable meaning for the undo button, but because in the case of group undo, there are several reasonable meanings. Choosing the meaning a user intends is impossible, and so it may sometimes be better not to guess.

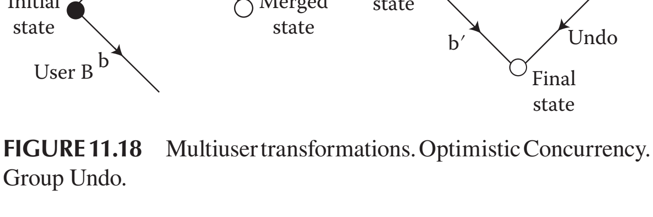


FIGURE 11.18 Multiuser transformations. Optimistic Concurrency. Group Undo.

11.6.2 Architectures for Networked Systems

Software architecture is about choosing what (in terms of code and functionality) goes where. For applications on a single machine these are often logical distinctions between parts of the code. For networked systems, "where" includes physical location, and the choice of location makes an enormous difference to the responsiveness of the system.

The simplest systems are almost always centralized client–server architectures, where the majority of computation and data storage happens in the central server. Many of the problems of race conditions and potential inconsistencies disappear. However, this means that every interaction requires a network interaction with the server meaning that feedback may be very slow. At the opposite extreme are replicated peer–peer architectures, where all the code is running on users' own PCs and the PCs communicate directly with one another. Feedback can now be instantaneous, but the complexity of algorithms to maintain consistency, catch-up late joiners, and so on can be very complex. Most systems operate somewhere between these two extremes with a central "golden copy" of shared data, but with some portion of the data on individual PCs, mobile phones, or other devices in order to allow rapid feedback. A notable exception to this is peer–peer sharing networks such as Gnutella where the only central resource is a sort of switchboard to allow client programs to contact one another. The reason for this is largely legal as these are often used to share copyright media! One reason this works is that the data is static, "Yesterday" is the same now as when it was first recorded in 1965 … although given it has the greatest number of covers of any song, perhaps not the best example!

In web applications, options are constrained by the features allowed in HTML and web browsers. Note that even a web form is allowing some local interaction (filling in the form) as well as some centralized interaction (submitting it). Applets allow more interaction, but the security limitations mean that they can only talk back to the server where they originated. Thus true peer–peer architectures are impossible on the web, but can be emulated by chat servers and similar programs that relay messages between clients. For Intranets, it is easier to either configure browsers so that they accept applets as trusted and thus with greater network privileges, or include special plug-in components to perform more complicated actions.

Mobile systems have yet more issues as they need to be capable of managing disconnected operation (when they have no physical or wireless connection to the network). A number of mechanisms have been developed over the years to allow local caching of network data, especially web pages. As applications, such as web mail and Google docs, shifted more functionality into network applications, this became more urgent leading to first vendor-specific technology such as Google Gears (Google 2010) and now standardized methods in HTML5 for creating and accessing local data for offline use of "web" applications (W3C 2010a,b).

For native applications also, smartphones often have replicated architectures with major resynchronization when connected through cheap/fast connection. For example, the iPhone SDK includes mechanisms for third parties to add synchronizable data.

11.6.3 Supporting Infrastructure

In order to help manage networked applications, various types of supporting infrastructure are being developed.

11.6.3.1 Awareness Servers

These keep track of which users are accessing particular resources (e.g., visiting a particular web page) so that you can be kept informed as to whether others are "near" you in virtual space, or whether friends are online and active (Palfreyman and Rodden 1996; ICQ 2001; SUN Microsystems 2001).

11.6.3.2 Notification Servers

These serve a similar role for data allowing client programs to register an interest in particular pieces of shared data. When the data is modified or accessed, the interested parties are "notified." For example, you may be told when a web page has changed or when a new item has been added to a bulletin board. Some notification servers also manage the shared data (Patterson, Day, and Kucan 1996) while others are "pure," just managing the job of notification (Ramduny, Dix, and Rodden 1998).

11.6.3.3 Event/Messaging Systems

These allow different objects in a networked environment to send messages to one another in ways which are more convenient than that allowed by the raw network. For example, they may allow messages to be sent to objects based on location-independent names, so that objects do not have to know where each other are.

11.6.3.4 Resource Discovery

Systems such as the Java JINI framework and Universal Plug-and-Play allow devices to find out about other devices close to them, for example, the local printer, and configure themselves to work with one another. As ubiquitous and mobile devices multiply, this will become increasingly important.

11.6.3.5 Web Service and Remote Procedure Call

Web service frameworks such as SOAP (Simple Object Access Protocol) or XML (extensible markup language)–RPC or nonweb remote procedure mechanisms such as Java's RMI or common object request broker architecture (CORBA) allow applications on different machines to easily connect to one another even though they may be written in different programming languages and run on different operating systems. In all cases, the parameters or arguments have to be serialized into a binary or ASCII test stream to be sent. As Figure 11.19 shows these are designed to be read by machines, not people!

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV = "http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi = "http://www.w3.org/1999/XMLSchema-instance"
  xmlns:xsd = "http://www.w3.org/1999/XMLSchema">
   <SOAP-ENV:Header>
   </SOAP-ENV:Header>
   <SOAP-ENV:Body>
      <ns1:sayHelloTo
         xmlns:ns1 = "Lookup"
      SOAP-ENV: encodingStyle = "http://schemas.xmlsoap.org/soap/
 encoding/">
         <name xsi:type = "xsd:string">Hello World</name>
      </ns1:sayHelloTo>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

FIGURE 11.19 SOAP XML encoding of Lookup ("Hello World").

Because of the complexity of SOAP and other RPC mechanisms, many web APIs now use representational state transfer (REST)-based protocols (Fielding 2000). REST takes a more data-oriented view of intercomponent communications compared with a function/method paradigm of RPC. Instead of defining bespoke operations, REST uses the native HTTP methods GET, POST, PUT, and DELETE often in combination with standard HTTP authentication and content negotiation. REST-based APIs often make use of JavaScript Object Notation encoding of the return results, which was originally designed as a lightweight way to return results to JavaScript web pages (suing AJAX or similar techniques), but has now become common as a cross-language format (Crockford 2010).

11.7 HISTORY, PARADIGM SHIFT, AND FUTURES

11.7.1 History

Timeline—key events for the Internet:

1968—First proposal for ARPANET–military and government research contracted to Bolt, Beranek & Newman

1971—ARPANET enters regular use

1973–1974—Redesign of lower level protocols leads to TCP/IP

1983—Berkeley TCP/IP implementation for 4.2BSD—public domain code

1980s—Rapid growth of NSFNET–broad academic use

1990s—WWW and widespread public access to the Internet

2000—WAP on mobile phones web transcends the Internet

Given the anarchic image of the web, it is strange that the Internet began its development as a U.S. military project. The suitability of the Internet for distributed management and independent growth stems not from an egalitarian or anticentralist political agenda, but from the need to make the network resilient to nuclear attack with no single point of failure.

In the 1970s when the Internet was first developing, it and other networks were mainly targeted at connecting together large computers. It was during the 1980s, with the rise of personal computing that local networks began to become popular.

However, even before that point, very local networks at the lab-bench level had been developed to link laboratory equipment, for example, the IEEE488 designed originally to link Hewlett Packard's proprietary equipment and then becoming an international standard. Ethernet, too, began life in commercial development at Xerox before becoming the de facto standard for local networking.

Although it is technical features of the Internet (decentralized, resilient to failures, hardware independent) that have made it possible for it to grow, it is the web that has made it become part of popular consciousness. Just as strange as the Internet's metamorphosis from military standard to anarchic cult is the web's development: from medium of exchange of large high-energy physics data sets to e-commerce, home of alternative web pages and online sex!

11.7.2 Paradigm Shift

During the 1970s through to the mid-1990s, networking was a technical phenomenon, enabling many aspects of business and academic life, but with very little public impact. However, this has changed dramatically over the last five years and now we think of a networked society. The Internet and other network technologies, such as SMS text messages, are not only transforming society, but at a popular and cultural level defining an era.

International transport, telecommunications, and broadcasting had long before given rise to the term global village. However, it seems this was more a phrase waiting for a meaning. Until recently the global village was either parochial—telephoning those you already know, or sanitized—views of distant cultures through the eyes of the travel agent or television camera. It is only now that we see chat rooms and web home pages allowing new contacts and friendship around the world—or at least among those affluent to get Internet access.

Markets too have changed due to global networks. It is not only the transnational that can trade across the world, and even my father-in-law runs a thriving business selling antiques through eBay.

Marketing has also had to face a different form of cross-cultural issue. Although selling the same product, a hoarding in Karachi may well be different from an ads in a magazine in Kentucky, reflecting the different cultural concerns. Global availability of web pages changes all that. You have to create a message that appeals to all cultures—a tall order. Those who try to replicate the targeting of traditional media by having several country-specific websites may face new problems—the global access to even these country-specific pages means that the residents of Kentucky and Karachi can compare the different ads prepared for them, and in doing so see how the company views them and their cultures.

Economics drives so much of popular as well as business development of networked society. One of the most significant changes in the United Kingdom was because of the changes in charging models. In the United States, local calls had long been free and hence so were Internet connections to local points of presence. The costs of Internet access in the United Kingdom (and even more important the perception of the cost) held back widespread use for many years. The rise of free or fixed-charge unmetered access changed nearly overnight the acceptability and style of use. Internet access used to be like a lightening guerrilla attack into the web territory, quick in and out before the costs mounted up, but became a full-scale occupation.

The need for telecom companies across the world to recover large investments in wireless band franchises combined with use rather than connection-based charging made possible by GPRS and third-generation mobile services, is now enabling a similar growth in mobile access to global networked information and services.

In an article in 1998, I used the term PopuNET to refer to a change in society that was not yet there, but would come. PopuNET is characterized by network access: everywhere, everywhen by everyone

This pervasive, permanent, popular access is similar to the so-called Martini principle applied more recently to mobile networking—anytime, anyplace, anywhere. Of course, Martini never pretends to be anything but exclusive, so not surprisingly these differ on the popular dimension! "Anyplace, anywhere" does correspond to the pervasive "everywhere" and "anytime" to the permanent "everywhen." However, there is a subtle difference, especially between "anytime" and "everywhen." "Anytime" means that at any time you choose you can connect. "Everywhen" means that at all places and all times you are connected. When this happens, one ceases to think of "connectedness" and it simply becomes part of the backdrop of life.

The combination of high-bandwidth networks, changes in charging models and always-on mobile connectivity have brought the goal of "everywhen" closer. However, in even the most developed nations, move out of major cities and bandwidth and connectivity rapidly begin to fall. So for the urban smartphone user, PopuNET is here, but as a truly pervasive experience "everywhen" is still coming.

PopuNET demands new interfaces and products, not just putting web pages on TV screens or spreadsheets on fridge doors. What these new interfaces will be is still uncertain, but the explosion of apps on the iPhone and similar platforms such as JIL (2010) shows one direction.

In fact many applications effectively assume constant connectivity; for example, many of the help pages are delivered as web pages. This can lead to problems when disconnected—especially in the case when looking for help to get connected! However, it also means that documentation can be easily updated. Similarly, most software now includes automatic updates meaning that you have the latest security fixes, but with the downside that as "essential" security update of 200 Mb may take a long time if you do not have a broadband connection. We will increasingly see an erosion of the distinctions between network and computer, symbolized by the popularity of the NetBook.

11.7.3 Futures (Near)

It is dangerous to predict far into the future in an area as volatile as this. One development that is already underway, which will make a major impact on user interfaces, is short-range networking, which will enable various forms of wearable and ubiquitous networks. Another is the introduction of network appliances, which will make the home "alive" on the network.

We have considered network aspects of continuous media at length. The fact that the existing Internet TCP/IP protocols do not enable guaranteed quality-of-service will put severe limits on its ability to act as an infrastructure for services such as video-on-demand or video sharing between homes. The update to TCP/IP which has been under development for several years, IPv6, will allow prioritized traffic; it falls short of real guaranteed QoS (IPng 2001).

It seems that this is an impasse. One of the reasons that IPv6 has taken so long is not the technical difficulty, but backwards compatibility and the problems of uptake on the existing world-wide infrastructure. Although most computers and routers do have IPv6 support, it is rare to see it actually used, and consequently, evolutionary change is hard. However, revolutionary change is also hard: One cannot easily establish a new parallel international infrastructure overnight. Or perhaps one can.

Mobile-phone services started with an infrastructure designed for continuous voice and are, through a series of quite dramatic changes, moving this toward a fully mixed media/data service. And it is a global network. Furthermore, more and more nonweb-based Internet services are using HTTP, the web protocol, to talk to one another in order to be "firewall friendly." In principle, a device could be web connected without being Internet connected.

In earlier editions, I suggested that possibly this may mean that *n*th-generation mobile networks effectively supplant the Internet allowing both web and voice traffic, and maybe being used as the bases of new wire-based standards. However, predictions are dangerous and this was completely wrong. Although phones themselves do still make use of separate voice and data channels, the movement across voice and media on wired and Wi-Fi connections has been exactly the opposite with Voice over IP services, notably Skype acting as a replacement for telephones, and various forms of on-demand video sitting alongside or even replacing terrestrial or satellite television.

One challenge that is certainly emerging, and a safe prediction, is the rise of the mobile phone as the most common (or only) form of Internet connectivity (and indeed computing) across many parts of the world including much of China, India, and Africa. Indeed, by the time of the next edition of this book, it is likely that mobile-phone web-access will already have overtaken conventional computing.

Man businesses are looking toward the "Next Billion" in these emerging markets (NextBillion 2010), who so far have been technologically disenfranchised. HP Labs are expecting that these Next Billion customers, many of whom may be illiterate or use languages with non-Latin character sets, may never have used a standard computer, but will access computation and information solely through a mobile-phone handset; and (HPL 2010). The applications and patterns of use that have arisen in desktop environments or even for Western urban youth may not be the most appropriate for this new market (Dix and Subramanian 2010).

REFERENCES

All web links below and other related links and material at: http://www.hiraeth.com/alan/hbhci/network/

Abowd, G. D., and A. J. Dix. 1992. Giving undo attention. *Interact Comput* 4(3):317–42.

AlexaWSP. 2006. *Alexa Web Search Platform Service*. http://pages.alexa.com/prod_serv/web_search_platform.html (accessed March 13, 2006).

Amazon. 2010 *Amazon Mechanical Turk*. http://www.mturk.com/ (accessed May 01, 2010).

Androutsellis-Theotokis, S., and D. Spinellis. 2004. A survey of peer-to-peer content distribution technologies. *ACM Comput Surv* 36(4):335–71.

Anslow, M. 2009. 'How 'smart fridges' could slash UK CO2 emissions and help renewables,' guardian.co.uk, Tuesday 28th April 2009. (online) (accessed November 10, 2009). http://www.guardian.co.uk/environment/2009/apr/27/carbon-emissions-smart-fridges-environmentally-friendly-appliances

Apple. 2010. *Integrating Sync Services into Your Application*. http://developer.apple.com/macosx/syncservices.html (accessed May 01, 2010).

Arehart, C., N. Chidambaram, S. Guruprasad, A. Homer, R. Howell, S. Kasippillai et al. 2000. *Professional WAP*. Birmingham, UK: Wrox Press. The Wrox series includes some of the best programmer-level texts on web and related technology: http://www.wrox.com/

Arons, B. 1997. SpeechSkimmer: A system for interactively skimming recorded speech. *ACM Trans Comput Hum Interact (TOCHI)* 4(1):3–38.

Barnard, P., and J. May. 1995. Interactions with advanced graphical interfaces and the deployment of latent human knowledge. In *Eurographics Workshop on the Design, Specification and Verification of Interactive Systems*, ed. F. Paternò, 15–49. Berlin, Germany: Springer Verlag. More information about ICS at: http://www.mrc-cbu.cam.ac.uk/personal/phil.barnard/ics/

Bellotti, V. 1993. Design for privacy in ubiquitous computing environments. In *Proceedings of CSCW'93*, 77–92. New York: ACM Press.

Benford, S., R. Anastasi, M. Flintham, A. Drozd, A. Crabtree, C. Greenhalgh, N. Tandavanitj, M. Adams, and J. Row-Farr. 2003. Coping with uncertainty in a location-based game. *IEEE Pervasive Comput* 2(3):34–41.

Benford, S., J. Bowers, L. Fahlen, J. Mariani, and T. Rodden. 1994. Supporting cooperative work in virtual environments. *Comput J* 37(8):635–68.

Berlage, T., and M. Spenke. 1992. The GINA interaction recorder. In *Proceedings of the IFIP WG2.7 Working Conference on Engineering for Human–Computer Interaction, Ellivuori, Finland*, 69–80. Amsterdam: North-Holland.

Berners-Lee, T., J. Hendler, and O. Lassila. 2001. "The Semantic Web". *Scientific American Magazine*, 17th May 2001.

Bizer, C., T. Heath, and T. Berners-Lee. 2009. Linked data—the story so far. *Int J Semant Web Inf Syst Spec Issue Linked Data* 5(3):1–22.

Björk, S., J. Falk, R. Hansson, and P. Ljungstrand. 2001. Pirates!–using the physical world as a game board. In *Proceedings of Interact 2001*, ed. M. Hirose, 9–13. Amsterdam, The Netherlands: IOS Press.

Bluetooth. 2001. *Official Bluetooth SIG Website*. http://www.bluetooth.com/ (accessed March 13, 2006).

Blythe, M., and A. Monk. 2005. Net neighbours: Adapting HCI methods to cross the digital divide. *Interact Comput* 17:35–56.

Buxton, W., and T. Moran. 1990. EuroPARC's Integrated Interactive Intermedia Facility (IIIF): Early experiences. In *Multi-User Interfaces and Applications, Proceedings of IFIP WG8.4 Conference*, ed. S. Gibbs and A. A. Verrijn-Stuart, 11–34. Heraklion, Greece. Amsterdam: North-Holland.

Calder, M., M. Kolberg, E. H. Magill, and S. Reiff-Marganiec. 2003. Feature interaction: A critical review and considered forecast. *Comput Netw* 41(1):115–41. DOI = http://dx.doi.org/10.1016/S1389-1286(02)00352-3.

Campbell, A., and K. Nahrstedt, eds. 1997. *Building QoS into Distributed Systems*. Boston, MA: Kluwer.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

CASIDE. 2005. CASIDE Project (EP/C005589): Investigating cooperative applications in situated display environments. http://www.caside.lancs.ac.uk/ (accessed March 13, 2006).

CERN. 2001. European organisation for nuclear research. http://public.web.cern.ch/Public/ (accessed March 13, 2006).

Chalmers, M., M. Bell, B. Brown, M. Hall, S. Sherwood, and P. Tennent. 2005. Gaming on the edge: Using seams in ubicomp games. In *Proceedings of ACM Advances in Computer Entertainment (ACE05)*, 306–9. New York: ACM Press.

Clarke, D., A. Dix, D. Ramduny, and D. Trepess, eds. 1997. Workshop on time and the web, staffordshire university. http://www.hiraeth.com/conf/web97/papers/ (accessed March 13, 2006).

Crockford, D. 2010. Introducing JSON. http://json.org (accessed May 1, 2010).

DataPortability. 2010. DataPortability Project. http://www.dataportability.org/ (accessed May 01, 2010).

Denbigh, A. 2003. *The Teleworking Handbook: The Essential Guide to Working from Where You Want*. 4th ed. London, U.K.: Methuen Drama.

Dix, A. J. 1987. The myth of the infinitely fast machine. In *People and Computers III—Proceedings of HCI'87*, 215–28. Cambridge, U.K.: Cambridge University Press.

Dix, A. J. 1991. *Formal Methods for Interactive Systems*. New York: Academic Press.

Dix, A. J. 1994. Seven years on, the myth continues. Research Report RR9405, University of Huddersfield, Huddersfield, U. K. http://www.hcibook.com/alan/papers/myth95/7-years-on.html (accessed March 13, 2006).

Dix, A. J. 1995. Dynamic pointers and threads. *Collab Comput* 1(3):191–216.

Dix, A. J. 1996. Closing the loop: Modelling action, perception and information. In *Proceedings of AVI'96—Advanced Visual Interfaces, Gubbio, Italy*, ed. T. Catarci, M. F. Costabile, S. Levialdi, and G. Santucci, 20–8. New York: ACM Press.

Dix, A. 2001a. artefact + marketing = product. *Interfaces* 48:20–1. London, UK: BCS-HCI Group. http://www.hiraeth.com/alan/ebulletin/product-and-market/ (accessed March 13, 2006).

Dix, A. 2001b. Cyber-economies and the Real World. Keynote—South African Institute of Computer Scientists and Information Technologists Annual Conference, SAICSIT 2001, 25–28 September 2001. Pretoria, South Africa. http://www.hcibook.com/alan/papers/SAICSIT2001/ (accessed March 13, 2006).

Dix, A. 2005. The brain and the web—a quick backup in case of accidents. *Interfaces* 65:6–7.

Dix, A. 2009. Paths and patches: Patterns of geonosy and gnosis. In *Chapter 1 in Exploration of Space, Technology, and Spatiality: Interdisciplinary Perspectives*, ed. P. Turner, S. Turner, and E. Davenport, 1–16. Hershey, PA: Information Science Reference.

Dix, A., and S. A. Brewster. 1994. Causing trouble with buttons. In *Ancilliary Proceedings of HCI'94, Glasgow, Scotland*, ed. D. England. London, UK: BCS-HCI Group. http://www.hcibook .com/alan/papers/buttons94/ (accessed March 13, 2006).

Dix, A., J. Finlay, G. Abowd, and R. Beale. 1998. *Human–Computer Interaction*. 2nd ed. Englewood Cliffs, NJ: Prentice Hall, Inc.

Dix, A., A. Friday, B. Koleva, T. Rodden, H. Muller, C. Randell, and A. Steed. 2005. Managing multiple spaces. In *Space, Spatiality and Technologies*, ed. P. Turner and E. Davenport, 151–72. Dordrecht, NL: Springer.

Dix, A., G. Lepouras, A. Katifori, C. Vassilakis, T. Catarci, A. Poggi, Y. Ioannidis et al. 2010. From the web of data to a world of action. Special Issue on Exploring New Interaction Designs Made Possible by the Semantic Web. *J Web Semant* 8(4):394–408.

Dix, A., T. Rodden, N. Davies, J. Trevor, A. Friday, and K. Palfreyman. 2000. Exploiting space and location as a design framework for interactive mobile systems. *ACM Trans Comput Hum Interact (TOCHI)* 7(3):285–321.

Dix, A., and S. Subramanian. 2010. IT for sustainable growth. *J Technol Manage Growing Econ* 1(1):35–54.

Dodge, M., and R. Kitchin. 2001. *Atlas of Cyberspace*. Reading, MA: Addison Wesley. http://www.cybergeography.org/atlas/

Dourish, P., and V. Bellotti. 1992. Awareness and coordination in shared workspaces. In *Proceedings of CSCW'92*, 107–14. New York: ACM Press.

Eagle, N. 2009. txteagle: Mobile crowdsourcing. In *Internationalization, Design and Global Development (IDGD 2009). LNCS 5623*, 447–56. Berlin: Springer-Verlag.

Edwards, W. K., and T. Rodden. 2001. *Jini, Example by Example*. Upper Saddle River, NJ: SUN Microsystems Press.

Electrolux. 1999. Screenfridge. http://www.electrolux.com/screenfridge/ (accessed March 13, 2006).

Ellis, C. A., and S. J. Gibbs. 1989. Concurrency control in groupware systems. In *Proceedings of 1989 ACM SIGMOD International Conference on Management of Data, SIGMOD Record*, 18(2):399–407. New York: ACM.

FCC. 2006. Voice over Internet Protocol. Federal Communications Commission. http://www.fcc.gov/voip/ (accessed March 13, 2006).

Fielding, R. T. 2000. *Architectural Styles and the Design of Network-Based Software Architectures*. PhD Dissertation. Irvine, CA: University of California.

Frank G. Halasz, Thomas P. Moran, and Randall H. Trigg. 1986. Notecards in a nutshell. *SIGCHI Bull*. 18, 4 (May 1986), 45–52. DOI = 10.1145/1165387.30859, http://doi.acm.org/10.1145/1165387.30859.

FOAF. 2000. Introducing FOAF. dated early 2000 http://www.foaf-project.org/original-intro (accessed May 01, 2010).

Fogg, B. J., J. Marshall, O. Laraki, A. Osipovich, C. Varma, N. Fang et al. 2001. What makes web sites credible? A report on a large quantitative study. In *Proceedings of CHI2001*, Seattle, 2001. Also *CHI Lett* 3(1):61–8. New York: ACM Press.

Foster, I. 2000. Internet computing and the emerging grid. *Nature* WebMatters. http://www.nature.com/nature/webmatters/grid/grid.html (accessed March 13, 2006).

Foster, I., and C. Kesselman, eds. 1999. *The Grid: Blueprint for a New Computing Infrastructure*. San Francisco, CA: Morgan-Kaufmann.

Gaver, W. W., R. B. Smith, and T. O'Shea. 1991. Effective sounds in complex situations: The ARKola simulation. In *Reaching Through Technology—CHI'91 Conference Proceedings*, ed. S. P. Robertson, G. M. Olson, and J. S. Olson, 85–90. New York: ACM Press.

Gellersen, H.-W., M. Beigl, and H. Krull. 1999. The MediaCup: Awareness technology embedded in an everyday object. In *Handheld & Ubiquitous Computing*, Lecture Notes in Computer Science, ed. H.-W. Gellersen. Vol. 1707, 308–10. Berlin, Germany: Springer.

Google. 2010. Gears: Improving Your Web Browser. http://gears.google.com/ (accessed March 6, 2010).

Gray, W. D., B. E. John, and M. E. Atwood. 1992. The precis of project ernestine or an overview of a validation of goms. In *Striking a Balance, Proceedings of the CHI'92 Conference on Human Factors in Computing Systems*, ed. P. Bauersfeld, J. Bennett, and G. Lynch, 307–12. New York: ACM Press.

Greenhalgh, C. 1997. Analysing movement and world transitions in virtual reality tele-conferencing. In *Proceedings of ECSCW' 97*, ed. J. A. Hughes, W. Prinz, T. Rodden, and K. Schmidt, 313–28. Dordrecht, The Netherlands: Kluwer Academic Publishers.

GRID. 2001. GRID Forum home page. http://www.gridforum.org/ (accessed March 13, 2006).

Griffiths, E., and A. Dix. 2008. The Pharmacist and the EPS (Electronic Prescription Service). In *Workshop HCI for Medicine and Health Care (HCI4MED) at HCI 2008*. Liverpool. http://www.hcibook.com/alan/papers/HCI4MED08-Pharmacist-and-the-EPS/ (accessed 10 Sept. 2010).

Gutwin, C., S. Benford, J. Dyck, M. Fraser, I. Vaghi, and C. Greenhalgh. 2004. Revealing delay in collaborative environments. In *Proceedings of ACM Conference on Computer–Human Interaction (CHI'04)*, 503–10. New York: ACM Press.

Halasz, F., T. Moran, and R. Trigg. 1987. NoteCards in a nutshell. In *Proceedings of the CHI+GI*, 45–52. New York: ACM Press.

Harrison, C., Z. Yeo, and S. E. Hudson. 2010. Faster progress bars: Manipulating perceived duration with visual augmentations. In *Proceedings of the 28th Annual SIGCHI Conference on Human Factors in Computing Systems (Atlanta, Georgia, April 10–15, 2010). CHI'10*. New York: ACM.

Hildebrandt, M., A. Dix, and H. A. Meyer. 2004. Time design. In *CHI'04 Extended Abstracts on Human Factors in Computing Systems (Vienna, Austria, April 24–29, 2004). CHI'04*, eds. E. Dykstra-Erickson, and M. Tscheligi, 1737–8. New York: ACM. DOI = http://doi.acm.org/10.1145/985921.986208.

Hindus, D., S. D. Mainwaring, N. Leduc, A. E. Hagström, and O. Bayley. 2001. Casablanca: Designing social communication devices for the home. In *Proceedings of CHI 2001*, 325–32. New York: ACM Press.

Howard, S., and J. Fabre, eds. 1998. Temporal aspects of usability. Special issue of *Interact Comput* 11(1):1–105.

HPL. 2010. HPL India: Innovations for the Next Billion Customers. http://www.hpl.hp.com/research/hpl_india_next_billion_customers/ (accessed May 1, 2010).

Hughes, D., S. Gibson, J. Walkerdine, and G. Coulson. 2006. Is deviant behaviour the norm on P2P file-sharing networks? In *IEEE Distributed Systems Online, February 2006*, 7(2). http://dsonline.computer.org (accessed March 13, 2006).

ICQ. 2001. ICQ. http://www.icq.com/products/whatisicq.html (accessed March 13, 2006).

IEEE. 2001. IEEE 802.11 Working Group. http://www.ieee802.org/11/ (accessed March 13, 2006).

Ipng. 2001. IP Version 6 (IPv6). http://playground.sun.com/pub/ipng/html (accessed March 3, 2007).

ISO/IEC7498. 1994. Information technology—Open Systems Interconnection—Basic Reference Model: The Basic Model. International Standards Organisation. http://www.iso.org (accessed March 3, 2007).

JIL. 2010. Joint Innovation Lab. http://www.jil.org/ (accessed May 1, 2010).

John, B. E. 1990. Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information. In *Empowering People—Proceedings of CHI'90 Human Factors in Computer Systems*, ed. J. C. Chew and J. Whiteside, 107–15. New York: ACM Press.

Johnson, C. 1997. What's the web worth? The impact of retrieval delays on the value of distributed information. In *Workshop on Time and the Web*, ed. D. Clarke, A. Dix, D. Ramduny, and D. Trepess. Staffordshire University. http://www.hiraeth.com/conf/web97/papers/johnson.html (accessed March 13, 2006).

Johnson, C., and P. Gray, eds. 1996. Temporal aspects of usability, report of workshop in glasgow, June 1995. *SIGCHI Bull* 28(2):32–61. New York: ACM Press.

JPEG. 2001. Joint Photographic Experts Group home page. http://www.jpeg.org/public/jpeghomepage.htm (accessed March 3, 2007).

Kistler, J. J., and M. Satyanarayanan. 1992. Disconnected operation in the CODA file system. *ACM Trans Comput Syst* 10(1):3–25.

Konstan, J. 2004. Introduction to recommender systems: Algorithms and evaluation. *ACM Trans Inf Syst* 22(1):1–4. DOI = http://doi.acm.org/10.1145/963770.963771.

Lavery, D., A. Kilgourz, and P. Sykeso. 1994. Collaborative use of X-Windows applications in observational astronomy. In *People and Computers IX*, ed. G. Cockton, S. Draper, and G. Wier, 383–96. Cambridge, UK: Cambridge University Press.

Light, A., and I. Wakeman. 2001. Beyond the interface: Users' perceptions of interaction and audience on websites. In *Interfaces for the Active Web (Part 1)*, ed. D. Clarke and A. Dix, Special Issue of *Interact Comput* 13(3):401–26. Amsterdam, The Netherlands: Elsevier.

Lock, S., J. Allanson, and P. Phillips. 2000. User-driven design of a tangible awareness landscape. In *Proceedings of Symposium on Designing Interactive Systems*, ed. D. Boyarski and W. Kellogg, 434–40. New York: ACM Press.

Lock, S., P. Rayson, and J. Allanson. 2003. Personality engineering for emotional interactive avatars. In *Human Computer*

Interaction, Theory and practice (Part II), Volume 2 of the Proceedings of the 10th International Conference on Human–Computer Interaction, 503–7. Mahwah, NJ: Lawrence Erlbaum Associates.

Mariani, J. A., and T. Rodden. (1991). The impact of CSCW on database technology. In *Proceedings of ACM Conference on Computer Supported Cooperative Work (includes critique of 'transparency' in a CSCW setting)*. New York: ACM Press.

Mauve, M. 2000. Consistency in replicated continuous interactive media. In *Proceedings of CSCW'2000*, 181–90. New York: ACM Press.

McDaniel, S. E., and T. Brinck. 1997. Awareness in collaborative systems: A CHI 97 workshop (report). In *SIGCHI Bulletin*, 29(4). New York: ACM Press.

McManus, B. 1997. Compensatory actions for time delays. In *Workshop on Time and the Web*, ed. D. Clarke, A. Dix, D. Ramduny, and D. Trepess. Staffordshire University. http://www.hiraeth.com/conf/web97/papers/barbara.html (accessed March 13, 2006).

Millett, L. I., B. Friedman, and E. Felten. 2001. Cookies and web browser design: Toward informed consent online. In *Proceedings of CHI2001*, CHI Lett 3(1):46–52. New York: ACM Press.

Mitsopoulos, E. 2000. *A Principled Approach to the Design of Auditory Interaction in the Non-Visual User Interface*. DPhil Thesis. University of York, UK. http://www.cs.york.ac.uk/ftpdir/reports/YCST-2000-07.zip (accessed March 3, 2007).

Mozilla Labs. 2010. Weave: Personalized, rich experiences across the web. https://mozillalabs.com/weave/ (accessed May 1, 2010).

MPEG. 2001. Moving Picture Experts Group home page. http://www.cselt.it/mpeg/ (accessed March 3, 2007).

mqtt.org. 2009. MQ Telemetery Transport. http://mqtt.org/ (accessed November 10, 2009).

NextBillion. 2010. Next Billion. http://www.nextbillion.net/about (accessed May 1, 2010).

Olson, M., and S. Bly. 1991. The Portland experience: A report on a distributed research group. *Int J Man Mach Stud* 34(2):11–228.

O'Reilly, T. 2005. What is web 2.0: Design patterns and business models for the next generation of software. dated 30 Sept.2005. http://oreilly.com/web2/archive/what-is-web-20.html (accessed May 1, 2010).

Palfreyman, K., and T. Rodden. 1996. A protocol for user awareness on the world wide web. In *Proceedings of CSCW'96*, 130–9. New York: ACM Press.

Patterson, J. F., M. Day, and J. Kucan. 1996. Notification servers for synchronous groupware. In *Proceedings of CSCW'96*, 122–9. New York: ACM Press.

Pausch, R. 1991. Virtual reality on five dollars a day. In *CHI'91 Conference Proceedings*, ed. S. P. Robertson, G. M. Olson, and J. S. Olson, 265–70. Reading, MA: Addison Wesley.

Ramduny, D., and A. Dix. 1997. Why, what, where, when: Architectures for co-operative work on the WWW. In *Proceedings of HCI'97*, ed. H. Thimbleby, B. O'Connaill, and P. Thomas, 283–301. Berlin, Germany: Springer.

Ramduny, D., A. Dix, and T. Rodden. 1998. Getting to know: The design space for notification servers. In *Proceedings of CSCW'98*, 227–35. New York: ACM Press.

Resnick, P., and H. R. Varian, eds. 1997. Communications of the ACM Special Issue on Recommender Systems 40(3):56–89.

Riedl, J., and P. Dourish. 2005. Introduction to the special section on recommender systems. *ACM Trans Comput Hum Interact* 12(3):371–3. DOI = http://doi.acm.org/10.1145/1096737.1096738.

Rodden, T. 1996. Populating the application: A model of awareness for cooperative applications. In *Proceedings of the 1996 ACM Conference on Computer-Supported Cooperative Work (CSCW'96)*, ed. M. S. Ackerman, 87–96. New York: ACM Press.

Rohs, M., J. G. Sheridan, and R. Ballagas. 2004. Direct manipulation techniques for large displays using camera phones. In *Proceedings of 2nd International Symposium on Ubiquitous Computing Systems (UCS2004)*. http://www.equator.ac.uk/index.php/articles/113 (accessed March 3, 2007).

Roussel, N. 1999. Beyond webcams and videoconferencing: Informal video communication on the web. In *Proceedings of the Active Web*. http://www.hiraeth.com/conf/activeweb/ (accessed March 3, 2007).

Roussel, N. 2001. Exploring new uses of video with videoSpace. In *Proceedings of EHCI'01, the 8th IFIP Working Conference on Engineering for Human-Computer Interaction, Lecture Notes in Computer Science*, 73–90. Berlin, Germany: Springer-Verlag.

Sandor, O., C. Bogdan, and J. Bowers. 1997. Aether: An awareness engine for CSCW. In *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work (ECSCW'97)*, ed. J. Hughes, 221–36. Dordrecht, The Netherlands: Kluwer Academic.

Sangha, A. 2001. Legal Implications of Location Based Advertising. Interview for the WAP Group. http://www.thewapgroup.com/53762_1.DOC (accessed March 3, 2007).

Sarvas, R., E. Herrarte, A. Wilhelm, and M. Davis. 2004. Metadata creation system for mobile images. In *Proceedings of Second International Conference on Mobile Systems, Applications, and Services (MobiSys2004)*, 36–48. New York: ACM Press.

Schneier, B. 1996. *Applied Cryptography*. 2nd ed. New York: Wiley. This is the best single point for cryptography, encryption, authentication, digital signatures, including full algorithms and source code on CD-ROM.

Semacode. 2006. Semacode. http://semacode.org/ (accessed March 3, 2007).

SETI@home. 2001. SETI@home—the search for extra-terrestrial intelligence. http://setiathome.ssl.berkeley.edu/ (accessed March 3, 2007).

Shneiderman, B. 1984. Response time and display rate in human performance with computers. *ACM Comput Surv* 16(3):265–86. New York: ACM Press.

Siegal, D. 1999. *Futurize Your Enterprise*. New York: Wiley.

Skype. 2006. Skype. http://www.skype.com/ (accessed March 3, 2007).

Stefik, M., D. G. Bobrow, G. Foster, S. Lanning, and D. Tatar. 1987. WYSIWIS revisited: Early experiences with multiuser interfaces. *ACM Trans Office Inf Syst* 5(2):147–67.

Stevens, W. R. 1998. *UNIX Network Programming, Volume 1, Networking APIs: Sockets and XTI*. Englewood Cliffs, NJ: Prentice Hall.

Stevens, W. R. 1999. *UNIX Network Programming, Volume 2, Interprocess Communications*. 2nd ed. Englewood Cliffs, NJ: Prentice Hall. W. Richard Stevens' books are classics on Internet protocols and programming. http://www.kohala.com/start/

Stifelman, L., B. Arons, and C. Schmandt. 2001. The Audio Notebook: Paper and pen interaction with structured speech. In *Proceedings of CHI2001*, CHI Lett 3(1):182–9. New York: ACM Press.

Sun, C., and C. Ellis. 1998. Operational transformation in real-time group editors: Issues, algorithms, and achievements. In *Proceedings of CSCW'98*, 59–68. New York: ACM Press.

SUN Microsystems. 2001. Awarenex. http://www.sun.com/research/features/awarenex/ (accessed March 3, 2007).

Taylor, A., S. Izadi, L. Swan, R. Harper, and W. Buxton. 2006. Building Bowls for Miscellaneous Media. Physicality 2006 workshop, Lancaster University, 6–7 Feb. 2006. http://www.physicality.org/ (accessed March 13, 2006).

thePooch. 2006. ThePooch. http://www.thepooch.com/ (accessed March 3, 2007).

Tillotson, J. 2005. 'Scent Whisper,' Central Saint Martins College of Art & Design. Exhibited in SIGGRAPH CyberFashion 0100, 2005. http://psymbiote.org/cyfash/2005/ (accessed March 3, 2007).

Toye, E., A. Madhavapeddy, R. Sharp, D. Scott, A. Blackwell, and E. Upton. 2004. Using Camera-phones to Interact with Context-Aware Mobile Services, Technical Report UCAM-CL-TR-609, University of Cambridge, Computer Laboratory. Retrieved 3 March 2007 from http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-609.pdf (accessed March 3, 2007).

txteagle. 2009. Empowering the largest knowledge workforce on Earth. http://txteagle.com/ (accessed July 2009).

Vidot, N., M. Cart, J. Ferriz, and M. Suleiman. 2000. Copies convergence in a distributed real-time collaborative environment. In *Proceedings of CSCW'2000*, 171–80. New York: ACM Press.

von Ahn, L., B. Maurer, C. McMillen, D. Abraham, and M. Blum. 2008. reCAPTCHA: Human-based character recognition via web security measures. *Science* 321:1465–8.

W3C. 2010a. Web SQL Database (Editors Draft). http://dev.w3.org/html5/webdatabase/ (accessed March 4, 2010).

W3C. 2010b. Web Storage (Editors Draft). dated 4th March 2010. http://dev.w3.org/html5/webstorage/ (accessed March 4, 2010).

Wang, D., and A. Mah. 2010. Google Wave Operational Transformation. http://www.waveprotocol.org/whitepapers/operational-transform (accessed May 1, 2010).

Wikipedia contributors. 2006. File sharing. Wikipedia, The Free Encyclopedia. http://en.wikipedia.org/w/index.php?title=File_sharing&oldid=34285993 (accessed March 3, 2007).

WiMedia. 2006. WiMedia Alliance. http://wimedia.org/ (accessed March 3, 2007).

Wisneski, G., H. Ishii, A. Dahley, M. Gorbet, S. Brave, B. Ullmer, and P. Yarin. 1998. Ambient display: Turning architectural space into an interface between people and digital information. In *Proceedings of the First International Workshop on Cooperative Buildings (CoBuild'98)*, Lecture Notes in *Comput Sci* 1370:22–32. Heidelberg: Springer-Verlag.

Witten, I. H., H. W. Thimbleby, G. Coulouris, and S. Greenberg. 1991. Liveware: A new approach to sharing data in social networks. *Int J Man Mach Stud* 34:337–48.

Xia, S., D. Sun, C. Sun, D. Chen, and H. Shen. 2004. Leveraging single-user applications for multi-user collaboration: The CoWord approach. In *Proceedings of ACM 2004 Conference on Computer Supported Cooperative Work*, 162–71. New York: ACM Press.

Zigbee. 2006. ZigBee Alliance. http://zigbee.org/ (accessed March 3, 2007).

Zimmerman, T. G. 1996. Personal area networks: Near-field intrabody communication. *IBM Syst J* 35(3&4):609–17.

12 Wearable Computers

Daniel Siewiorek, Asim Smailagic, and Thad Starner

CONTENTS

12.1 INTRODUCTION

Computers have become a primary tool for office workers, allowing them to access the information they need to perform their jobs; however, accessing information is more difficult for mobile users. With current computer interfaces, the user must focus both physically and mentally on the computing device instead of the environs. In a mobile environment, such interfaces may interfere with the user's primary task. However, many mobile tasks could benefit from computer support. (Smailagic and Siewiorek 1993; Siewiorek et al. 1998; Lyons and Starner 2001.) Our focus is the design of wearable computers that augment, instead of interfere, with the user's tasks. Carnegie Mellon University's (CMU) VuMan 3 project provides an example of how the introduction of wearable computing to a task can reap many rewards.

12.2 MAINTENANCE INSPECTION USING A VUMAN 3 WEARABLE COMPUTER

Many maintenance activities begin with an inspection in which problems are identified. Job orders and repair instructions are generated from the results of the inspection. The VuMan 3 wearable computer was designed for streamlining Limited Technical Inspections (LTI) of amphibious tractors for the U.S. Marines at Camp Pendleton, California (Smailagic et al. 1998). The LTI is a 600-element, 50-page checklist that usually takes 4–6 hours to complete. The inspection includes an item for each part of the vehicle (e.g., front left track, rear axle, windshield wipers, etc.). VuMan 3 created an electronic version of this checklist. The system's interface was arranged as a menu hierarchy and a physical dial, and selection buttons controlled navigation. The top level consisted of a menu that gave a choice of function. Once the inspection function was chosen, the component being inspected was selected by its location on the vehicle. At each stage, the user could go up one level of the hierarchy.

The inspector selects one of four possible options about the status of the item: (1) serviceable, (2) unserviceable, (3) missing, or (4) on equipment repair order. Further explanatory comments about the item can be selected (e.g., the part is unserviceable due to four missing bolts).

The LTI checklist consists of a number of sections, with approximately 100 items in each section. The user sequences through each item by using the dial to select "next item," or "next field." A "smart cursor" helps automate some of the navigation by positioning the user at the next most likely action.

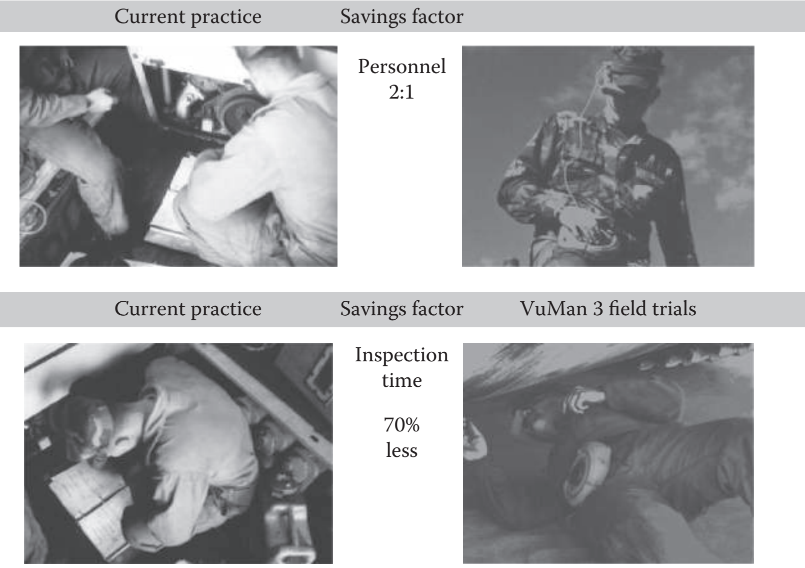


FIGURE 12.1 VuMan 3 savings factors.

As part of the design process, a field study was performed. In typical troubleshooting tasks, one marine would read the maintenance manual to a second marine who performs the inspection. With the VuMan 3, only one marine is needed for the task as he has the electronic maintenance manual with him. Thus, the physical manual does not have to be carried into hard-to-reach places.

The most unanticipated result was a 40% reduction in inspection time. The bottom right image of Figure 12.1 demonstrates the reason for this result. Here, the marine is on his side looking up at the bottom of the amphibious tractor. In such places, it is hard to read or write on the clipboard typically used for inspections. The marine constantly gets into position, crawls out to read instructions, crawls back into position for the inspection, and then crawls out again to record the results. In addition, the marine tends to do one task at a time when he might have five things he has to inspect in one place. This extra motion has a major impact on the time required to do a task. By making information truly portable, wearable computers can improve the efficiency of this application and many other similar ones.

The second form of time savings with the VuMan 3 occurred when the inspection is finished. The wearable computer requires a couple of minutes to upload its data to the logistics computer. The manual process, however, required a typist to enter the marine's handwritten text into the computer. Given that the soldier may have written his notes in cold weather while wearing gloves, the writing may require some interpretation. This manual process represents another 30% of the time.

Such redundant data entry is common when users are mobile (Starner et al. 2004). There are numerous checklist-based applications including plant operations, preflight checkout of aircraft, inventory, and so on, that may benefit from a form-filling application run on a wearable computer. In the case of the VuMan 3 project, the results were striking. From the time the inspection was started until the data was entered into the logistics computer, 70% of the time was saved by using the wearable. There was a potential saving by reducing maintenance crews from two to one. Finally, there was also a savings in weight over paper manuals.

12.3 FACTORS IN WEARABLE COMPUTER DESIGN

Designing wearable computer interfaces requires attention to many different factors due to their closeness to the body and their use while performing other tasks. For the purposes of discussion, we have created the "CAMP" framework, which consists of the following factors:

Corporal: Wearables should be designed to interface physically with the user without discomfort or distraction.

Attention: Interfaces should be designed for the user's divided attention between the physical and virtual worlds.

Manipulation: When mobile users lose some of the dexterity assumed by desktop interfaces, controls should be quick to find and simple to manipulate.

Perception: A user's ability to perceive displays, both visual and audio, is also reduced while mobile. Displays should be simple, distinct, and quick to navigate.

Power, heat, on-body and off-body networking, privacy, and many other factors also affect on-body computing (Starner 2001). Many of these topics are the subjects of current research, and much work will be required to examine how these factors interrelate. Due to space constraint, we will concentrate mainly on CAMP principles and practice in the remainder of this chapter.

12.4 CORPORAL: DESIGN GUIDES FOR WEARABILITY

The term “wearable” implies the use of the human body as a support environment for the object described. Society has historically evolved its tools and products into more portable, mobile, and wearable form factors. Clocks, radios, and telephones are examples of this trend. Computers are undergoing a similar evolution. Simply shrinking computing tools from the desktop paradigm to a more portable scale does not take advantage of a whole new context of use. While it is possible to miniaturize keyboards, human evolution has not kept pace by shrinking our fingers. There is no Moore’s Law for humans. The human anatomy introduces minimal and maximal dimensions that define the shape of wearable objects, and the mobile context also defines dynamic interactions. Conventional methods of interaction, including keyboard, mouse, joystick, and monitor, have mostly assumed a fixed physical relationship between user and device. With wearable computers, the user’s physical context may be constantly changing. Symbol’s development of a wearable computer for shipping hubs provides an example of how computing must be adapted for the human body.

As a company, Symbol is well known for its bar code technology. However, it is also one of the first successful wearable computer companies, having sold over 100,000 units from its WSS 1000 line of wearable computers (see Figure 12.2). The WS-1000 consists of a wrist-mounted wearable computer that features a laser bar code scanner encapsulated in a ring worn on the user’s finger. This configuration allows the user to scan bar codes while keeping both hands free to manipulate the item being scanned. Because the user no longer has to fumble with a desk-tethered scanner, these devices increase the speed at which the user can manipulate packages and decrease the overall strain on the user’s body. Such features are important in shipping hubs, where millions of packages are scanned by hand every year. Symbol spent over US$5 million and devoted 40,000 hours of testing to develop this new class of device, and one of the major challenges was adapting the computer technology to the needs of the human body (Stein et al. 1998).

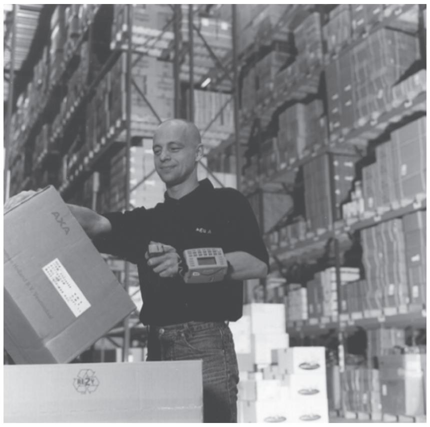


FIGURE 12.2 Symbol’s WSS 1000 series wrist-mounted wearable computer with ring scanner.

One of the first observations made was that users may be of widely varying shapes and sizes. Specifically, Symbol’s scanner had to fit the fingers of both large men and small women. Similarly, the wrist unit had to be mounted on both large and small wrists. Even though the system’s wires were designed to be unobtrusive, the system must be designed to break away if entangled and subjected to strain. This policy provided a safeguard for the user.

Initial testing discovered other needs that were obvious in hindsight. For example, the system was strapped to the user’s forearm while the user exerted himself in moving boxes. Soon, the “soft-good” materials, which were designed for the comfort of the user, became sodden with sweat. After one shift, the user was expected to pass the computer to the operator on the next shift. Not only was the sweat-laden computer mount considered “gross,” it also presented a possible health risk. This problem was solved by separating the computer mount from the computer itself. Each user received his own mount, which he could keep adjusted to his own needs. After each shift, the computer could be removed from the user’s mount and placed in the replacement user’s mount.

Another unexpected discovery is that the users tended to use the computer as body armor. When a shipping box would begin to fall on the user, the user would block the box with the computer mounted on his forearm, as that was the least sensitive part of his body. Symbol’s designers were surprised to see users adapt their work practices to use the rigid forearm computer to force boxes into position. Accordingly, the computer’s case was designed out of high-impact materials. However, another surprise came with longer term testing of the computer.

Employees in the test company’s shipping hubs constantly reached into wooden crates to remove boxes. As they reached into the crates, the computer would grind along the side. After extended use, holes would appear in the computer’s casing, eventually damaging the circuitry. Changing the composition of the casing to be resistant to both abrasion and impact finally fixed the problem.

After several design cycles, Symbol presented the finished system to new employees in a shipping hub. After a couple of weeks’ work, test results showed that the new employees felt the system was cumbersome, whereas established employees who had participated in the design of the project felt that the wearable computer provided a considerable improvement over the old system of package scanning. After consideration, Symbol’s engineers realized that these new employees had no experiential basis for comparing the new system to the past requirements of the job. As employees in shipping hubs are often short-term, a new group of employees were recruited. For two weeks, these employees were taught their job using the old system of package scanning: the employee would reach into a crate, grasp a package, transfer it to a table, grasp a handheld scanner, scan the package, replace the scanner, grasp the package, and transfer it to its appropriate conveyer belt. The employees were then introduced to the forearm-mounted WS-1000. With the wearable computer, the employee would squeeze his index and middle finger together to trigger the ring-mounted scanner to scan the package while reaching for it, grasp the package, and transfer it to the appropriate conveyer belt in one fluid motion. These employees returned very positive scores for the wearable computer.

This lesson—that perceived value and comfort of a wearable computer is relative—was also investigated by Bodine and Gemperle (2003). In short interviews, users were fitted with a backpack or armband "wearable" and told that the system was either a police monitoring device (similar to those used for house arrest), a medical device for monitoring health, or a device for use during parties. The subjects were then asked to rate the devices on various scales of desirability and comfort. Not surprisingly, the police "wearable" was considered the least desirable. However, the police function elicited more negative *physical* comfort ratings, and the medical function elicited more positive physical comfort ratings even though they were the same device. In other words, perceived comfort can be affected by the supposed function of the device.

Researchers have also explored wearability in more general terms. Wearability is defined as the interaction between the human body and the wearable object. Dynamic wearability includes the human body in motion. Design for wearability considers the physical shape of objects and their active relationship with the human form. Gemperle et al. explored history and cultures including topics such as clothing, costumes, protective wearables, and carried devices (Gemperle et al. 1998; Siewiorek 2002). They studied physiology, biomechanics, and the movements of modern dancers and athletes. Drawing upon the experience of CMU's wearable's group over two dozen generations of machines representing over a hundred person years of research, they codified the results into guidelines for designing wearable systems. These results are summarized in Table 12.1.

This team also developed a set of wearable forms to demonstrate how wearable computers might be mounted on the body. Each of the forms was developed by applying design guidelines and followed a simple pattern for ensuring wearability. The pods were designed to house electronic components. All of the forms are between 3/8 in. and 1 in. thick, and flexible circuits can fit comfortably into the 1/4 in.-thick flex zones. Beginning with acceptable areas and the humanistic form language, the team considered human movement in each individual area. Each area is unique, and some study of the muscle and bone structure was required along with common movements. Perception of size was studied for each individual area. For testing, minimal amounts of spandex was stretched around the body to attach the forms. The results are shown in Figure 12.3.

These studies and guidelines provide a starting point for wearable systems designers. However, there is much work to be done in this area. Weight was not considered in these

TABLE 12.1
Design for Wearability Attributes

Attribute	Comments
Placement	Identify where the computer should be placed on the body. Issues include identifying areas of similar size across a population, areas of low movement/flexibility, and areas large in surface area.
Humanistic form language	The form of the object should work with the dynamic human form to ensure a comfortable fit. Principles include inside surface concave to fit body, outside surface convex to deflect objects, tapering sides to stabilize form on body, and radiusing edges/corners to provide soft form.
Human movement	Many elements make up a single human movement: mechanics of joints, shifting of flesh, and the flexing and extending of muscles and tendons beneath the skin. Allowing for freedom of movement can be accomplished in one of two ways: (1) by designing around the more active areas of the joints or (2) by creating spaces on the wearable form into which the body can move.
Human perception of size	The brain perceives an aura around the body. Forms should stay within the wearer's intimate space, so that perceptually they become a part of the body. The intimate space is between zero and 5 in. off the body and varies with position on the body.
Size variations	Wearables must be designed to fit many types of users. Allowing for size variations is achieved in two ways: (1) static anthropometric data, which details point-to-point distances on different sized bodies and (2) consideration of human muscle and fat growth in three dimensions using solid rigid areas coupled with flexible areas.
Attachment	Comfortable attachment of forms can be created by wrapping the form around the body, rather than using single-point fastening systems such as clips or shoulder straps.
Contents	The system must have sufficient volume to house electronics, batteries, and so on, that, in turn, constrains the outer form.
Weight	The weight of a wearable should not hinder the body's movement or balance. The bulk of the wearable object weight should be close to the center of gravity of the human body minimizing the weight that spreads to the extremities.
Accessibility	Before purchasing a wearable system, walk and move with the wearable object to test its comfort and accessibility.
Interaction	Passive and active sensory interaction with the wearable should be simple and intuitive.
Thermal	The body needs to breathe and is very sensitive to products that create, focus, or trap heat.
Esthetics	Culture and context will dictate shapes, materials, textures, and colors that perceptually fit the user and their environment.

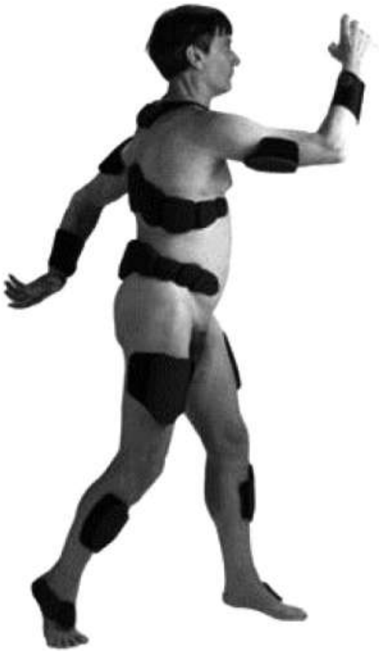


FIGURE 12.3 Forms studied for wearability.

studies, nor was the long-term physiological effects such systems might have on the wearer's body. Similarly, fashion can affect the perception of comfort and desirability of a wearable component. As wearable systems become more common and are used for longer periods of time, it will be important to test these components of wearability.

12.5 ATTENTION

Humans have a finite and nonincreasing capacity that limits the number of concurrent activities they can perform. Herb Simon observed that human effectiveness is reduced as they try to multiplex more activities. Frequent interruptions require a refocusing of attention. After each refocus of attention, a period of time is required to reestablish the context prior to the interruption. In addition human short-term memory can hold seven plus or minus two (i.e., five to nine) chunks of information. With this limited capacity, today's systems can overwhelm users with data, leading to information overload. The challenge to human computer interaction design is to use advances in technology to preserve human attention and to avoid information saturation.

In the mobile context, the user's attention is divided between the computing task and the activities in the physical environs. Some interfaces, like some augmented realities (Azuma 1997) and Dual Purpose Speech (Lyons et al. 2004), try to integrate the computing task with the user's behavior in the physical world. The VuMan 3 interface did not tightly couple the virtual and real worlds, but the computer interface was designed specifically for the user's task and allowed the user to switch rapidly between a virtual interface and his hands-on vehicle inspection.

However, office productivity tasks, such as e-mail or web searching, have little relation to the user's environment. The mobile user must continually assess what attentional resources he can commit to the interface and for how long before switching attention back to his primary task. Oulasvirta et al. specifically examine such situations (Oulasvirta et al. 2005) by fitting cameras to mobile phones and observing users attempting web search tasks while following predescribed routes. Subjects performed these tasks in a laboratory, in a subway car, riding a bus, waiting at a subway station, walking on a quiet street, riding an escalator, eating at a cafeteria and conversing, and navigating a busy street. Web pages required an average of 16.2 seconds to load and had considerable variance, requiring the user to attend the interface. The subjects shifted their attention from the phone interface more often depending on the task: 35% of page loadings in the laboratory versus 80% of the page loadings while walking a quiet street. The duration of continuous attention on the mobile device also varied depending on the physical environment: 8–16 seconds for the laboratory and cafe versus below 6 seconds for riding the escalator or navigating a busy street. Similarly, the number of attention switches depending on the demands of the environment.

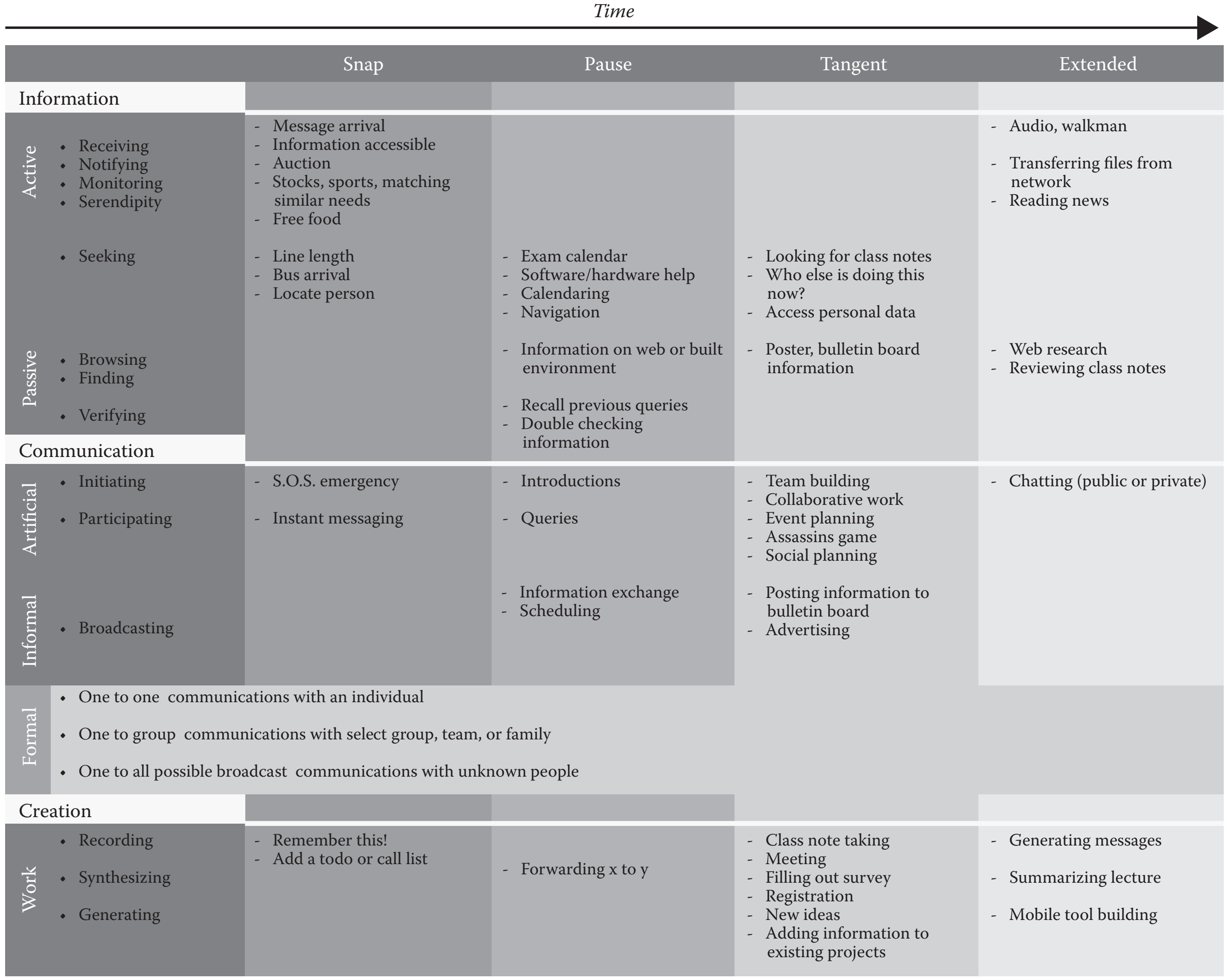
The authors note that even riding an escalator requires demands on attention (e.g., choosing a correct standing position, monitoring personal space for passersby, and determining when the end is in order to step off). Accordingly, they are working on a "Resource Competition Framework," based on the Multiple Resource Theory of attention (Wickens and Hollands 1999), to relate mobile task demands to the user's cognitive resources. This framework helps predict when the mobile user will need to adopt attentional strategies to cope with the demands of a mobile task. The authors report four such strategies that were observed in their study. (1) *Calibrating attention* refers to the process where the mobile user first attends to the environment and determines the amount of attention he needs to devote to the environment versus the interface. (2) *Brief sampling over long intervals* refers to the practice of only attending to the environment in occasional brief bursts to monitor for changes that may require a deviation from plan, such as reading while walking an empty street. (3) *Task finalization* refers to subjects' preference to finish, when sufficiently close, a task or subtask before switching attention back to the physical environment. (4) *Turntaking capture* occurs when the user is conversing with another person. Attending and responding to another person requires significant concentration, leading to minimal or no attention to the mobile interface.

The third author who has been using his wearable computer to take notes on his everyday life since 1993 has remarked on similar strategies in his interactions. Describing these attentional strategies and designing interfaces that leverage them will be important in future mobile interfaces. Much research has been performed on aircraft and automobile cockpit design to design interfaces that augment but do not interfere with the pilot's primary task of navigating the vehicle. However, only recently has it begun to be possible to instrument mobile users and examine interface use (and misuse) "in-the-field" for the mobile computer user. Now, theories of attention can be applied and tested to everyday life situations.

This newfound ability to monitor mobile workers may help us determine how *not* to design interfaces. In contrast to the VuMan 3 success described earlier, Ockerman's PhD thesis "Task Guidance and Procedure Context: Aiding Workers in Appropriate Procedure Following" warns that mobile interfaces, if not properly designed, may hinder the user's primary task (Ockerman 2000). Ockerman studied experienced pilots inspecting their small aircraft before flying. When a wearable computer was introduced as an aid to completing the aircraft's safety inspection checklist, the expert pilots touched the aircraft less (a way many pilots develop an intuition as to the aircraft's condition). In addition, the pilots relied too much on the wearable computer system, which was purposely designed to neglect certain safety steps. The pilots trusted the wearable computer checklist to be complete instead of relying on their own mental checklists. Ockerman shows how such interfaces might be improved by providing context for each step in the procedure. Another approach would be integrating the aircraft itself into the interface (e.g., use augmented reality to overlay graphics on the aircraft indicating where the pilot physically inspects the plane).

Most recently, the Defense Advanced Research Project Agency's Augmented Cognition project (Kollmorgen et al. 2005) aims to create mobile systems that monitor their user's attentional resources and records or delays incoming information to present it to the user in a more orderly and digestible time sequence. These systems exploit mobile electroencephalogram readings or functional near infrared imaging to monitor the user's brain activations and relate these results to the user's current state (Archinoetics). Such projects, if successful on a larger scale, could reveal much about the mental resources required for truly mobile computing (Archinoetics site).

The Attention Matrix, shown in Figure 12.4, (Anhalt et al. 2001) categorizes activities by the amount of attention they require. The activities are Information, Communication, and Creation. Individual activities are categorized by the amount



Time

		Snap	Pause	Tangent	Extended
Information					
Active	• Receiving • Notifying • Monitoring • Serendipity	- Message arrival - Information accessible - Auction - Stocks, sports, matching similar needs - Free food			- Audio, walkman - Transferring files from network - Reading news
	• Seeking	- Line length - Bus arrival - Locate person	- Exam calendar - Software/hardware help - Calendaring - Navigation	- Looking for class notes - Who else is doing this now? - Access personal data	
Passive	• Browsing • Finding		- Information on web or built environment	- Poster, bulletin board information	- Web research - Reviewing class notes
	• Verifying		- Recall previous queries - Double checking information		
Communication					
Artificial	• Initiating	- S.O.S. emergency	- Introductions	- Team building - Collaborative work - Event planning - Assassins game - Social planning	- Chatting (public or private)
	• Participating	- Instant messaging	- Queries		
Informal	• Broadcasting		- Information exchange - Scheduling	- Posting information to bulletin board - Advertising	
Formal	• One to one communications with an individual • One to group communications with select group, team, or family • One to all possible broadcast communications with unknown people				
Creation					
Work	• Recording • Synthesizing • Generating	- Remember this! - Add a todo or call list	- Forwarding x to y	- Class note taking - Meeting - Filling out survey - Registration - New ideas - Adding information to existing projects	- Generating messages - Summarizing lecture - Mobile tool building

FIGURE 12.4 Attention matrix.

of distraction they introduce in units of increasing time: (1) Snap, (2) Pause, (3) Tangent, and (4) Extended. The Snap duration is an activity that is usually completed in a few seconds, such as checking your watch for the time. The user should not have to interrupt his primary activity to perform this activity. The Pause action requires the user to stop current activity, switch to the new but related activity, and then return to the previous task within a few minutes. Pulling over to the side of the road and checking directions is an example of a pause. A Tangent action is a medium length task that is unrelated to the action that the user is engaged in. Receiving an unrelated phone call is an example of a tangent activity. An Extended action is when the user deliberately switches his task, beginning a wholly new long-term activity. For the car driver, stopping at a motel and resting for the night is an extended activity.

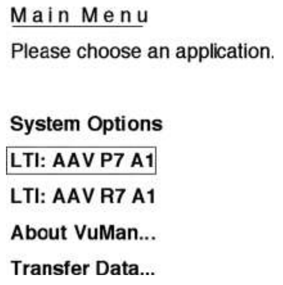
As distractions on the left of the matrix take less time from the user's primary activity, our intent is to move activities of the matrix toward the left side (Snap). Our goal is to evaluate how this process extends to a larger sample of applications.

12.6 MANIPULATION

12.6.1 VuMan 3 Dials Pointing

VuMan 3 added a novel manipulation interface suitable for use when physical attention is occupied. The VuMan 3 has a low-resolution display and, consequently, a purely textual interface. Figure 12.5 shows a sample screen from the user interface. The user navigates through a geographically organized hierarchy: top, bottom, front, rear; then left, right, and more detail. Eventually, at the node leafs, individual components are identified. There are over 600 of these components. Each component is indicated to be "serviceable" or "unserviceable." If it is serviceable, then no further information is given. If it is unserviceable, then one of a small list of reasons is the next screen.

The user can return up the hierarchy by choosing the category name in the upper right corner, or sequence to the next selection in an ordering of the components. Once a component is marked as serviceable or unserviceable, the next selection in the sequence is automatically displayed for the user. Furthermore, each component has a probability associated with it of being serviceable, and the cursor is positioned over the most likely response for that component.


Main Menu
Please choose an application.

System Options

LTI: AAV P7 A1

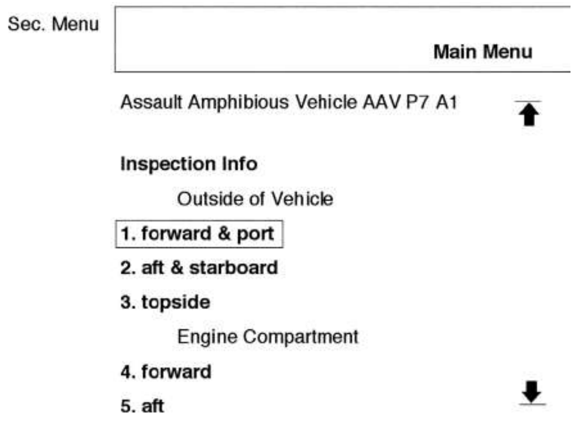
LTI: AAV R7 A1

About VuMan...

Transfer Data...

FIGURE 12.5 VuMan 3 options screen.

The screen contained navigational information. Sometimes there is more on a logical screen than can fit on a physical screen. Screen navigation icons are on the left-hand side of the screen. The user can go to the previous physical screen or next physical screen that is a functional part of the logical screen. The user can always go back to the Main Menu. In Figure 12.6, the VuMan 3 information screen is shown, with different sections to inspect. The inspection is divided into sections, and different people can be inspecting different sections in parallel. The inspector would pick a section, highlight it by rotating the dial, and then select the highlighted item by pressing a button. The inspector would then receive a detailed set of instructions on what to do. In Figure 12.7, the inspector is instructed to check for damage and bare metal. The "smart cursor" anticipates that the inspector will be filling in the "status field" whose current value is "none." By clicking, a list of options is displayed, the first of which is "serviceable." With the marine LTI, the item is serviceable in 80% of the cases. By ordering the most probable selection first, the interface emulates a paper checklist where most of the items will be checked as "OK." The smart cursor then assumes the most likely step. There is no need to even move the dial—you merely need to click on the highlighted option. For example, in Figure 12.8 "serviceable" has been filled in and the box signifying the next activity is


Sec. Menu

Main Menu

Assault Amphibious Vehicle AAV P7 A1

Inspection Info

Outside of Vehicle

1. forward & port

2. aft & starboard

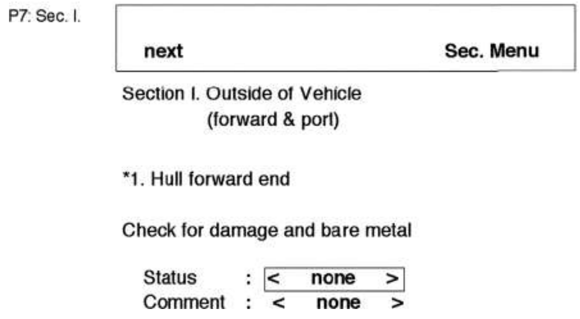
3. topside

Engine Compartment

4. forward

5. aft

FIGURE 12.6 VuMan 3 information screen.


P7: Sec. I.

next **Sec. Menu**

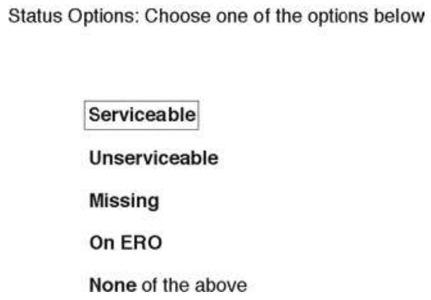
Section I. Outside of Vehicle
(forward & port)

*1. Hull forward end

Check for damage and bare metal

Status : < none >
Comment : < none >

FIGURE 12.7 VuMan 3 hull forward screen.



Status Options: Choose one of the options below

Serviceable

Unserviceable

Missing

On ERO

None of the above

FIGURE 12.8 VuMan 3 status screen.

"next." If all entries are "serviceable," one would simply tap the button multiple times. If an item is "unserviceable," the dial is turned and "unserviceable" is selected. Next, a list of reasons why that particular device was unserviceable would appear. The dial is rotated and one or more of the options are selected. Since more than one reason may be selected as to why it is unserviceable, "done" is selected to indicate completion. The selected items would appear in the "comment" field. When the checklist is completed, the data is uploaded to the logistics computer, which would then generate the job work orders.

Several lessons were derived from building the system. As part of the design cycle, a mouse (essentially a disk with buttons) was tested. However, the physical configuration of the device could be ambiguous. Was the left button in the proper position when the mouse's tail was toward the user or away from the user? Were the buttons supposed to be at the top? The dial removed this orientation ambiguity.

Another design lesson was to minimize cables. An earlier system had a cable connecting the battery, a cable for the mouse, and a cable for the display. These wires quickly became knotted. To avoid this problem, the VuMan 3 design used internal batteries, and the dial was built into the housing. The only remaining wire was the one to the display.

A third lesson was that wearable computers have a minimum footprint that is comfortable for your hand. While the keyboards of palmtop computers are getting smaller, evolution has not correspondingly shrunk our fingers. The thickness of the electronics will become thinner. Eventually, it will be as thick as a sheet of plastic or incorporated into clothing. However, there will be a minimal footprint for the interface. Furthermore, the interface—no matter where it is located on your body—is operated in the same way. This is a major feature of the dial. It can be worn on your hip or in the small of the back. In airplane manufacturing, where workers navigate small spaces, the hip defines the smallest diameter through which the person can enter. Here a shoulder holster is preferred for the wearable computer.

The marines' oversized coverall pockets were an advantage for the system. The soldiers could drop the computer into their coveralls and operate it through the cloth of the pocket. In terms of simplicity, as well as orientation independence, the dial integrated with the presentation of information on the screen. Everything on the screen could be considered to be on a circular list. In most cases, there are less than a dozen items on a screen that are selectable. This sparse screen is an advantage on a head-mounted display where the user may be reading while moving. The font must be large enough to read while the screen is bouncing. The dial should be an intuitive interface for web browsing. Probably, there are less than a half dozen items on a typical page to select, and it is rotated clockwise or counter clockwise. A button is then used to select the highlighted item. VuMan 3 had three types of buttons that all performed the same function. The buttons support left-hand and right-hand thumb-dominant as well as a central for finger-dominant users.

12.6.2 Mobile Keyboards

The VuMan 3 addressed the problem of menu selection in the mobile domain and effectively used a 1D dial to create a pointing device that can be used in many different mobile domains. However, for tasks like wireless messaging, more free-form text entry is needed. Although speech technology has made great strides in the last decade, speech recognition (SR) is very difficult in the mobile environment and often suffers from high error rates. In addition, speech is often not socially acceptable (in hospitals, meetings, classrooms, etc.). Keyboard interfaces still provide one of the most accurate and reliable methods of text entry.

Since 2000, wireless messaging has been creating billions of dollars of revenue for mobile phone service providers, and over 1 trillion messages are currently being typed per year. Until recently, many of these messages were created using the Multitap or T9 input method on phone keypads. Yet studies have shown that users average a slow 10–20 words per minute (wpm) using these common typing methods (for comparison, a highly skilled secretary on a desktop averages 70–90 wpm). Given the obvious desire for mobile text input, Human–Computer Interaction (HCI) researchers have begun reexamining keyboards. While keyboard entry has been well studied in the past, mobility suggests intriguing possibilities. For example, if an adequate method of typing can be combined with a sufficient display for the mobile market, computing may move "off-the-desktop" permanently.

Traditionally, text entry studies emphasize learnability, speed, and accuracy. However, a mobile user may not be able to devote all of his attention to the text entry process. For example, he may be taking notes on a conversation and wish to maintain eye contact with his conversational partner. Or, he may be in a meeting and may hide his keyboard under the desk to avoid distracting others with the keyboard's noise and the motion of his fingers. The user might also attempt to enter text while walking and need to attend his physical environment instead of looking at the screen. These conditions all describe "blind" typing where the user enters text with only occasional glances at the screen to ensure that the text has been entered correctly.

Lyons et al. and Clawson et al. (2005) have performed longitudinal studies on two keyboards, Handykey's Twiddler (Figure 12.9) and the mini-QWERTY "thumb" keyboard (Figure 12.10), to determine if they might achieve desktop

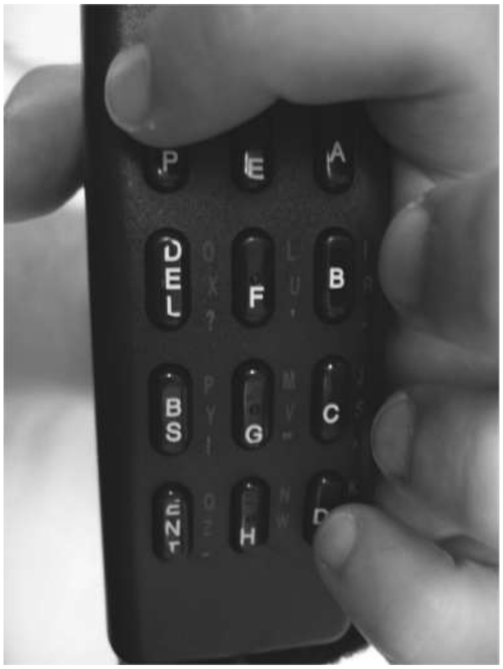


FIGURE 12.9 Handykey's Twiddler.

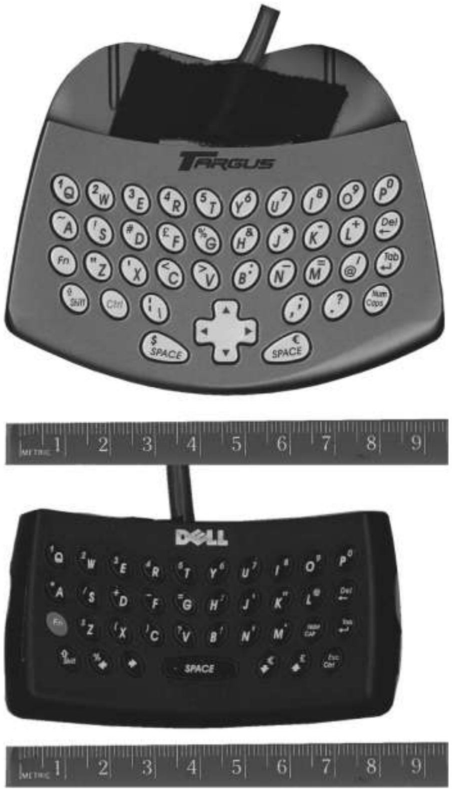


FIGURE 12.10 Mini-QWERTY thumb keyboard.

level text entry in the mobile domain. As the "average" desktop entry rate was considered to be 30 wpm, including hunt-and-peck typists, this benchmark was chosen as the minimum for speed. Traditionally, very high accuracy is desired for desktop typing. However, as a culture of informal e-mail and SMS messaging has developed, less accurate typing has become common. The community is debating how to reconcile speed and accuracy measures; however, error rates of approximately 5% per character are common in current mobile keyboard studies.

With the Twiddler, novices averaged 4 wpm during the first 20-minute session and averaged 47 wpm after 25 hours of practice (75 twenty-minute sessions). The fastest user averaged 67 wpm, which is approximately the speed of one of the authors who has been using the Twiddler for 12 years. While 25 hours of practice seems extreme, a normal high school typing class involves almost three times that training time to achieve a goal of 40 wpm.

Even so, mobile computer users may already have experience with desktop-QWERTY keyboards. Due to their familiarity with the key layout, these users might more readily adopt a mini-QWERTY keyboard for mobile use. Can a mini-QWERTY keyboard achieve desktop rates? The study performed by Clawson et al., examined the speed and accuracy of experienced desktop typists on two different mini-QWERTY designs. These subjects averaged 30 wpm during the first 20-minute session and increased to 60 wpm by the end of 400 minutes of practice!

While both of these studies easily achieved desktop typing rates and had error rates comparable to past studies, can these keyboards be used while mobile? While neither study tested keyboard use while the user was walking or riding in a car, both experimented with blind text entry (in that, in at least one condition, typists could not look at the keyboard nor the output of their typing). When there was a statistically significant difference between blind and normal typing conditions, experienced Twiddler typist slightly improved their speeds and decreased their error rates. However, experienced mini-QWERTY typists were significantly inhibited by the blind condition, with speeds of 46 wpm and approximately three times the error rate even after 100 minutes of practice. These results might be expected in that Twiddler users who are trained to type without visual feedback from the keyboard whereas the mini-QWERTY keyboard design assumes that the user can see the keyboard to help disambiguate the horizontal rows of small keys.

The results of these studies demonstrate that there are multiple ways that desktop typing rates can be achieved on a mobile device. The question remains, however, whether the benefits of typing quickly while "blind" or moving will be sufficient to cause users to learn a new text entry method. Other benefits might also affect the adoption of keyboards in the future. For example, a 12-button device like the Twiddler can be the size of a small mobile phone and still perform well, whereas 40-button mini-QWERTY keyboards may have already shrunk as much as possible for users' hands. Another factor may be adoption of mobile computing in developing countries. According to Techweb, almost 1 billon mobile phones were shipped in 2005. Many new mobile phone users will not have learned to type on a Roman alphabet keyboard and may be more concerned with quick learning than compatibility with desktop input skills.

12.7 SPEECH INTERFACES

12.7.1 Vocollect

Mobile keyboards are not suitable for applications in which hands-free control is necessary, such as warehouse applications. Pittsburgh-based Vocollect focuses on package manipulation—in particular, the warehouse-picking problem. In this scenario, a customer places an order consisting of several different items stored in a supplier's warehouse. The order transmits from the warehouse's computer to an employee's wearable computer (Figure 12.11). In turn, each item and its location are spoken to the employee through a pair of headphones. The employee can control how this list is announced through feedback via SR and can also report inventory errors as they occur. The employee accumulates the customer's order from the warehouse's shelves and ships it. This audio-only interface also frees the employee to manipulate packages with both hands, whereas a pen-based system would be considerably more awkward. As of December 2000, Vocollect had approximately 15,000 users and revenues between US$10 and US$25 million.

12.7.2 Navigator Wearable Computer with Speech Input

Boeing has been pioneering "augmented reality" using a head-mounted, see-through display. As the user looks at the aircraft, the next manufacturing step is superimposed on the appropriate portion of the aircraft. One of their first applications is fabrication of wire harnesses. Every aircraft is essentially unique. They may be from different airlines. Even if they are from the same airline, one might be configured for a long haul route and another for a short haul route. The airline may specify different configurations. For example, their galleys will be in different places, the wire harnesses would change, and so on. Wire harnesses are fabricated months before they are assembled into the aircraft. The assembly worker starts with a peg board measuring about 3 ft. high and 6 ft. long. Mounted on the board is a full-sized diagram of the final wire harness. Pegs provide support for the bundles of wire as they form. The worker selects a precut wire, reads its identification number, looks up the wire number on a paper list to find the starting coordinates of the wire, searches for the wire on the diagram, and threads the wire following the route on the diagram. With augmented reality, the worker selects a wire and reads the wire identification from the bar code. A head tracker provides the computer with information on where the worker is looking and superimposes the route for that particular wire on the board. Trial evaluations indicate a savings of 25% of the assembly effort primarily due to the elimination of cross-referencing the wire with paper lists.

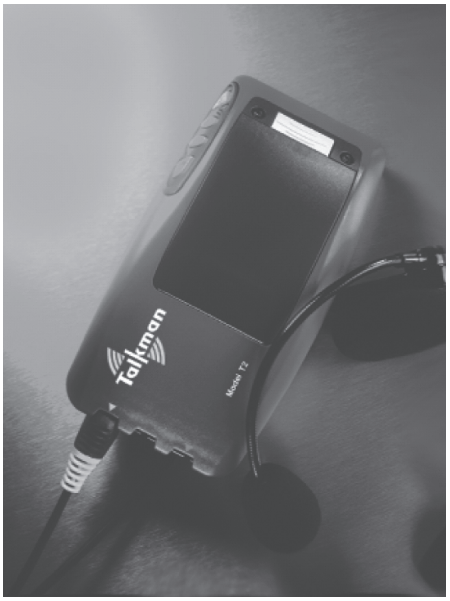


FIGURE 12.11 Vocollect's audio-based wearable computer.

The Navigator 2, circa 1995, is designed for a voice-controlled aircraft inspection application (Siewiorek, Smailagic, and Lee 1994; Smailagic and Siewiorek 1994; Smailagic and Siewiorek 1996). The SR system, with a secondary manually controlled cursor, offers complete control over the application in a hands-free manner, allowing the operator to perform an inspection with minimal interference from the wearable system. Entire or portions of aircraft manuals can be brought on-site as needed, using wireless communication. The results of inspection can be downloaded to a maintenance logistic computer.

Consider one portion of Navigator 2's application, three-dimensional inspections. The application was developed for McClellan Air Force Base in Sacramento, California and the KC-135 aerial refueling tankers. Every 5 years these aircraft are stripped down to bare metal. The inspectors use magnifying glasses and pocket knives to hunt for corrosion and cracks.

At start-up, as shown in Figure 12.12, the application prompts the user for either their choice of activating the SR system or not. The user then proceeds to the Main Menu. From this location, several options are available, including online documentation, assistance, and the inspection task (Figure 12.13). Once the user chooses to begin an inspection, information about the inspection is entered, an aircraft type to examine is selected, and the field of interest is narrowed from major features (Left Wing, Right Tail, etc.) (Figure 12.14) to more specific details (individual panes in the cockpit window glass) (Figure 12.15). A coordinate system is superimposed on the inspection region. The horizontal coordinates begin from the nose and the vertical coordinates are "water lines" derived as if the airplane was floating. The inspector records each imperfection in the skin at the corresponding location on the display. The area covered by each defect as well as the type of defect, such as corroded, cracked, or missing, are recorded. To maximize usability, each item or control may be selected simply by speaking its name. Figure 12.16 shows the Navigator 2 systems in use.

The user navigates to the display corresponding to the portion of the skin currently being inspected. This navigation is partially textual based on buttons (choose aircraft type to be inspected) and partially graphical based on side

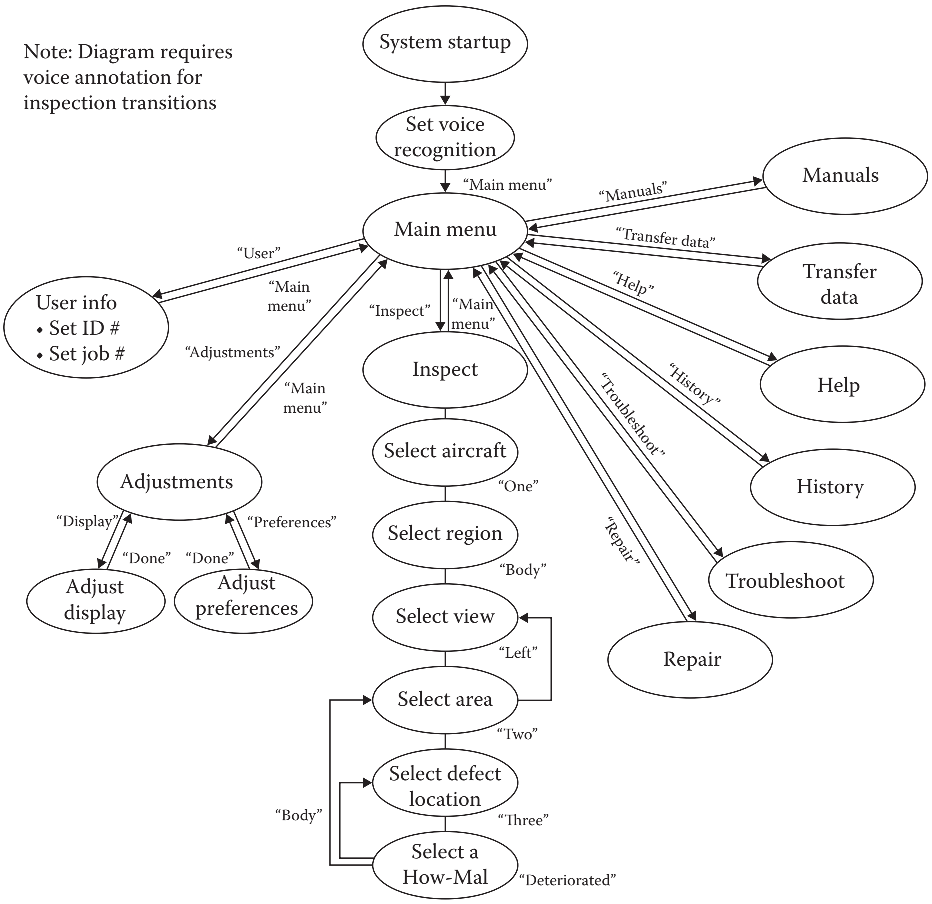


FIGURE 12.12 Navigator 2 state machine.

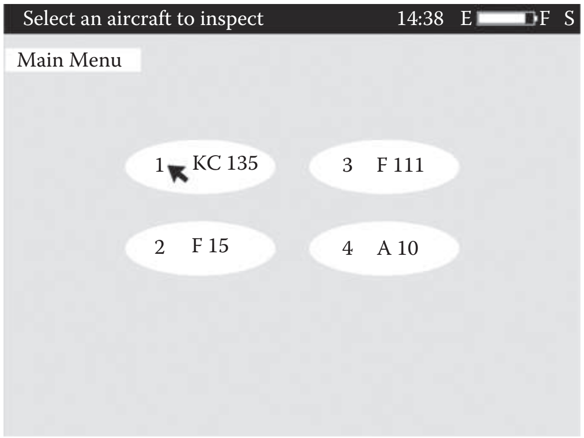


FIGURE 12.13 Navigator 2 main menu.

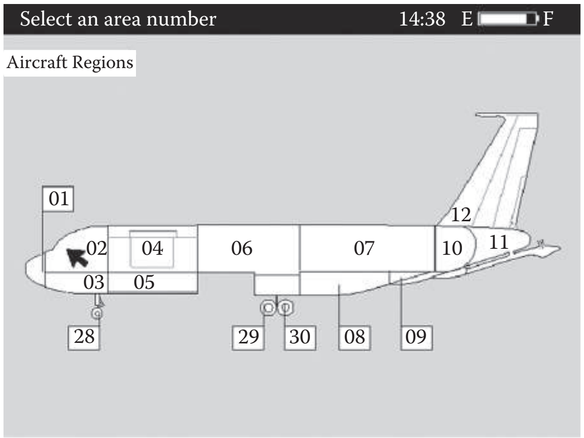


FIGURE 12.14 KC 135 inspection regions.

perspectives of the aircraft (choose area of aircraft currently being inspected). The navigation can be performed either through a joystick input device or through the use of speech input. The speech input is exactly the text that would be selected. The positioning of the imperfection is done solely through the joystick since speech is not well suited for the pointing necessary to indicate the position of the imperfection. As the cursor is moved by the joystick, the coordinates and the type of material represented by the cursor are displayed at the bottom of the screen. If a defect is at the current position, a click produces a list of reasons why that material would be defective such as corrosion, scratch, and so on. The defect type can be selected by the joystick or by speaking its name and the information would go into the database. The user can navigate to the main selection screen by selecting the "Main Menu" option on all of the screens. One level up in the hierarchy can also be achieved through a single selection.

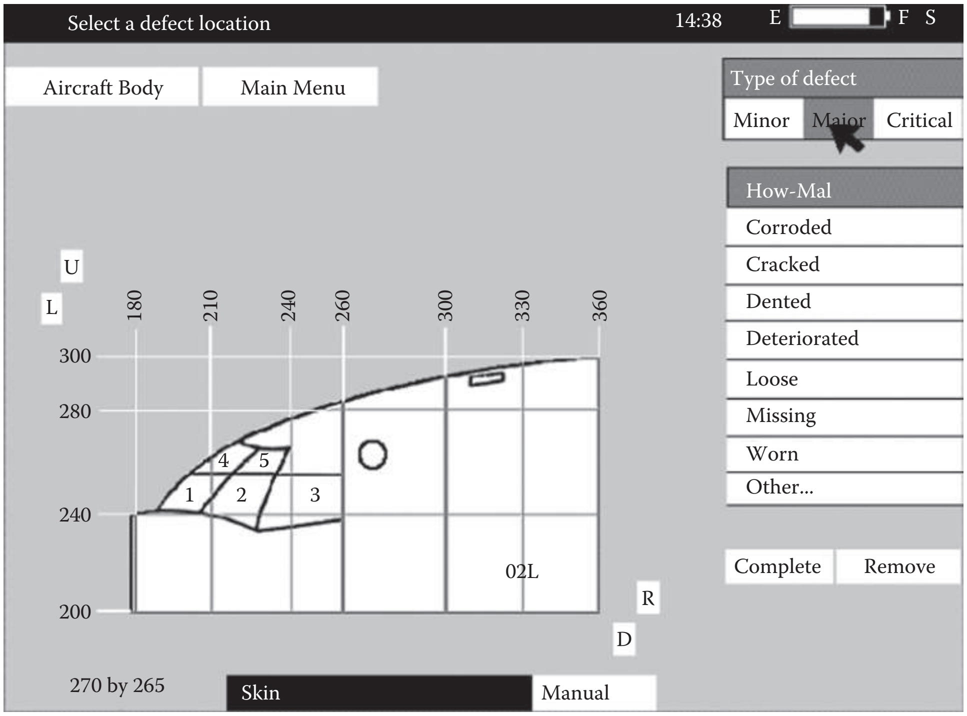


FIGURE 12.15 Locating and labeling a defect type.

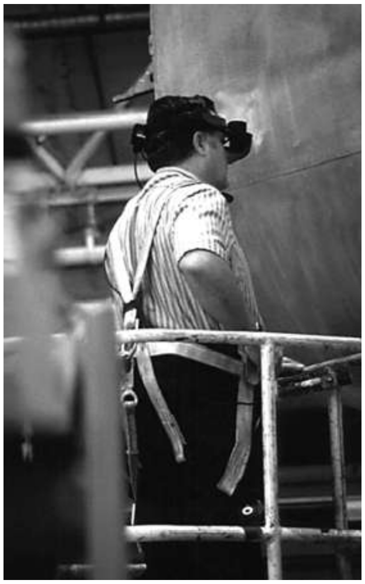


FIGURE 12.16 Inspector entering information into Navigator 2 system.

The relationship between the user interface design principles and the Navigator 2 user interface is as follows:

- Simplicity of function: The only functions available to the user are to enter skin imperfections for one of four aircrafts, to transfer data to another computer, to enter identification information both for the vehicle and for the inspector, and to see a screen that describes the Navigator 2 project.
- No textual input: The identification information required entering numbers. A special dialogue was developed to enable the entering of numeric information using the joystick as an input device. This was cumbersome for the users but only needed to be performed once per inspection.
- Controlled navigation: The interface was arranged as a hierarchy. The top level consisted of a menu that gave a choice of function. Once the inspection function and then the vehicle were chosen, the area of the skin inspected was navigated to via selecting an area of the aircraft to expand. Once an imperfection was indicated, the user had to select one of the allowable types of imperfections. At each stage, the user could go up one level of the hierarchy or return to the Main Menu.

One of the lessons learned with Navigator 2 is the power of forcing the use of a common vocabulary. Since the average age of the aircraft is 35 years, the type of defects encountered is a very stable set. Previously, one inspector would call a defect “gouged” whereas another inspector would call the same defect a “scratch.” What is the difference between a gouge and a scratch? How much material does it take? How much time does it take to repair? What skill of labor is needed? The logistics problem is much more difficult without a standardized vocabulary. Thus, there is a serendipitous advantage in injecting more technology.

A second lesson is that in some cases the SR front end mistakenly produces the wrong output. SR systems typically have an error rate of 2%–10%. The unexpected output may cause the application to produce the wrong result. In one of Navigator 2’s early demonstrations, the user was attempting to exit the application. The SR system thought a number was spoken. At that point, the application was expecting a second number, but there was no match since the user was saying "exit," "quit," "bye," and so on. The system appeared to be frozen when in actuality there was a mismatch between what the application software was expecting and what the user thought the application state was. The solution was to give the user more feedback on the state of the application by additional on-screen clues. Also, a novel application input test generator was developed that took a description of the interface screens and created a list of all possible legal exits from each screen.

A third lesson learned was the criticality of response time. When SR was done in software on Navigator (circa 1995), it was 12 times real time, which became very frustrating. People are less patient when they are on the move than when they are at a desktop. People at a desk are willing to wait 3 minutes for the operating system to boot up, but when you are on the move, expectations are for instant response like that of portable tools such as a flashlight. For example, some airplanes have a digital computer to control the passengers' overhead lights. It is disconcerting that when the button is pushed, it may take 2 or 3 seconds before the light turns on. Even a couple of second delay in a handheld device is disruptive. Users typically continue to push buttons until there is a response. The extra inputs cause disconnect between the software and the user. The software receives a stream of inputs, but the user sees outputs that are related to inputs given a long time before the screen appears. The situation is similar to listening to yourself talk when there is a second or two delay in the sound played back. The user easily becomes very confused.

The field evaluation indicated that the inspection is composed of three phases. The inspectors would spend the same amount of time maneuvering their cherry picker to access a region of the airplane, visually inspecting and feeling the airplane's skin, and recording the defect's type and location. Navigator 2 reduced the paperwork time by half resulting in an overall time savings of about 18%. Training time to familiarize inspectors with the use of Navigator 2 was about 5 minutes after which they would proceed with actual inspections. A major goal of field evaluations is that users perform productive work. They do not want to redo something that was already done once.

The typical inspection requires about 36 hours discovering approximately 100 defects. Today, the inspector takes notes on a clipboard. Upon completion, the inspector fills out forms on a computer. Each defect takes 2–3 minutes to enter. The data entry is thus an additional 3- to 4-hour task. Navigator 2 transmits the results of the inspection by radio in less than 2 minutes.

In summary, evaluations of inspectors before and after the introduction of Navigator 2 indicated a 50% reduction in the time to record inspection information (for an overall reduction of 18% in inspection time) and almost two orders of magnitude reduction in time to enter inspection information into the logistics computer (from over 3 hours to 2 minutes). In addition, Navigator 2 weighs 2 lb. compared with the cart the inspectors currently use with 25 lb. of manuals.

12.8 SPEECH TRANSLATION

The Speech Recognition/Language Translation (SR/LT) application consists of three phases: (1) speech to text language recognition, (2) text to text LT, and (3) text to speech synthesis. The application running on Tactical Information Assistant-Prototype (TIA-P), circa 1996, is the Dragon Multilingual Interview System (MIS), jointly developed by Dragon Systems and the Naval Aerospace and Operational Medical Institute (NAOMI). It is a keyword-triggered multilingual playback system, which listens to a spoken phrase in English, proceeds through a SR front end, plays back the recognized phrase in English, and after some delay (~8–10 seconds) synthesizes the phrase in a foreign language (Croatian). The other, local person can answer with Yes, No, and some pointing gestures. The Dragon MIS has about 45,000 active phrases in the following domains: medical examination, mine fields, road checkpoints, and interrogation. Therefore, a key characteristic of this application is that it deals with a fixed set of phrases and includes one-way communication. A similar system is used in Iraq as a briefing aid to interrogate former Iraqi intelligence officials and to speak with civilians about information relevant to locating individuals. This shows the viability of the approach.

TIA-P is a commercially available system, developed by CMU, incorporating a 133-MHz 586 processor, 32-MB DRAM, 2-GB IDE Disk, full-duplex sound chip, and spread spectrum radio (2 Mbps, 2.4 GHz) in a ruggedized, handheld, pen-based system designed to support speech translation applications. TIA-P is shown in Figure 12.17.

Dragon loads into memory and stays memory resident. The translation uses uncompressed ~20 KB of .WAV files per phrase. There are two channels of output: the first plays in English and the second in Croatian. A stereo signal can be split and one channel directed to an earphone, and the second to a speaker. This is done in hardware attached to the external speaker. An Andrea noise canceling microphone is used with an on-off switch.

Speech translation for one language (Croatian) requires a total of 60-MB disk space. The SR requires an additional 20–30 MB of disk space.

TIA-P has been tested with the Dragon speech translation system in several foreign countries: Bosnia (Figure 12.18), Korea, and Guantanamo Bay, Cuba. TIA-P has also been used in human intelligence data collection and experimentation with the use of electronic maintenance manuals for F-16 maintenance.

The following lessons were learned during the TIA-P field tests: wires should be kept to a minimum; handheld display is convenient for checking the translated text; standard external electrical power should be available for use internationally; battery lifetime should be extended; ruggedness is important.

The smart modules (circa 1997) are a family of wearable computers dedicated to the speech processing application (Smailagic, Siewiorek, and Reilly 2001). A smart module provides a service almost instantaneously and is configurable

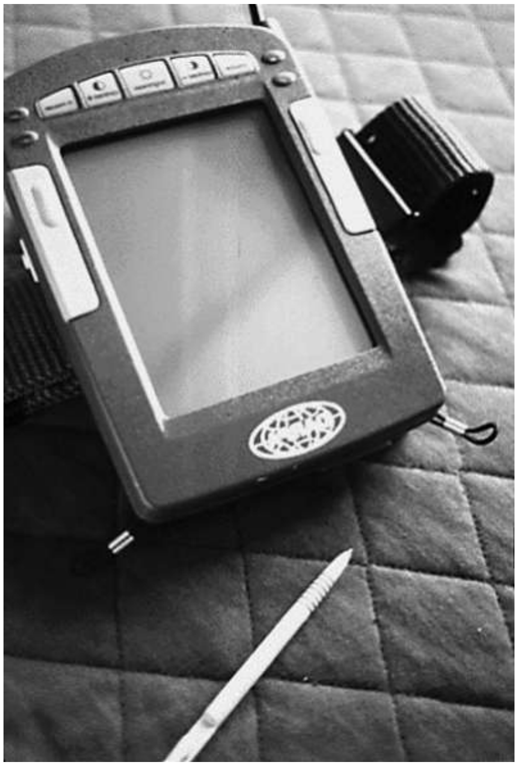


FIGURE 12.17 Tactical Information Assistant-Prototype wearable computer.

FIGURE 12.18 U.S. soldier in Balkans using Tactical Information Assistant-Prototype.

for different applications. The design goals also included: reduce latency, remove context swaps, and minimize weight, volume, and power consumption (Reilly 1998; Martin 1999; Smailagic 1997). The functional prototype consists of two functionally specialized modules, performing LT and SR. The first module incorporates speech to text language recognition and text to speech synthesis. The second module performs text to text LT. The LT module runs the PANLITE LT software (Frederking and Brown 1996), and the SR module runs CMU's Sphinx II continuous, speaker-independent SR software (Ravishankar 1996; Li et al. 1989) and Phonebox Speech Synthesis software.

Figure 12.19 depicts the structure of the speech translator from English to a foreign language and vice versa. The speech is input into the system through the SR subsystem. A user wears a microphone as an input device, and background noise is eliminated using filtering procedures. A language model, generated from a variety of audio recordings and data, provides guidance for the SR system by acting as a knowledge source about the language properties. The LT engine uses an example-based machine translation (EBMT) system, which takes individual sentence phrases and compares them to a corpus of examples it has in memory to find phases it knows how to translate. A lexical machine translation (glossary) translates any unknown word that may be left. The EBMT engine translates individual "chunks" of the sentence using the source language model and then combines them with a model of the target language to ensure correct syntax. When reading from the EBMT corpus, the system makes several random-access reads while searching for the appropriate phrase. Since random reads are done multiple times, instead of loading large, continuous chunks of the corpus into memory, the disk latency times will be far more important than the disk bandwidth. The speech generation subsystem performs text to speech conversion at the output stage. To make sure that misrecognized words are corrected, a Clarification Dialog takes place on-screen. It includes the option to speak the word again or to write it in. As indicated in Figure 12.19, an alternative input modality could be the text from the Optical Character Recognition subsystem (such as scanned documents in a foreign language), which is fed into the LT subsystem.

User-interface design went through several iterations based on feedback during field tests. The emphasis was on getting completely correct two-way speech translation, and having an easy to use, straightforward interface for the clarification dialogue.

The SR code was profiled and tuned. Profiling was performed to identify "hot spots" for hardware and software acceleration and to reduce the required computational and storage resources. A six times speedup was achieved over the original desktop PC system implementation of LT, and five times smaller memory requirements (Christakos 1998). Reducing operating system swapping and code optimization made a major impact. Input to the module is audio and output is ASCII text. The SR module is augmented with

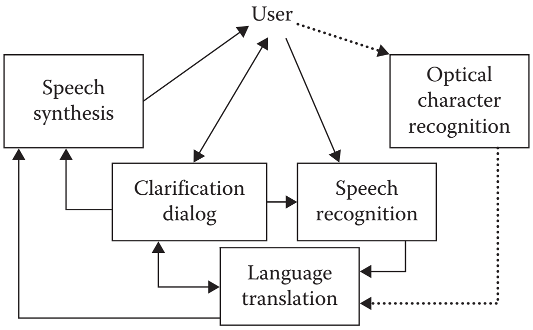


FIGURE 12.19 Speech translator system structure.

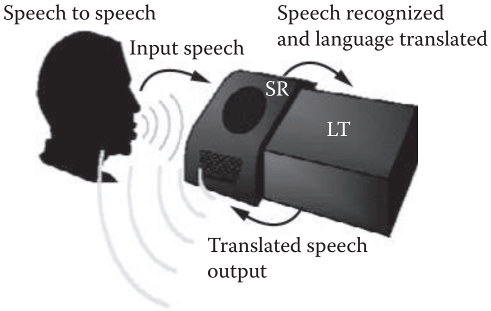


FIGURE 12.20 Speech recognizer (SR) and language translator (LT) smart module.

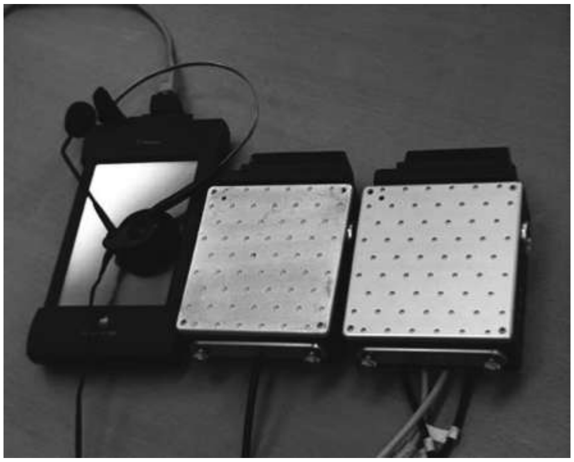


FIGURE 12.21 Speech translator smart module functional prototype.

speech synthesis. Figure 12.20 illustrates a combination of the LT module, and SR module, forming a complete standalone audio-based interactive dialogue system for speech translation.

Target languages included Serbo-Croatian, Korean, Creole French, and Arabic. Average LT performance was 1 second per sentence.

The key factors that determine how many processes can be run on a module are memory, storage space, and available CPU cycles. To minimize latency, the entirety of an application's working dataset should be able to stay memory resident.

Figure 12.21 depicts the functional prototype of the speech translator smart module, with one module performing LT, and another one SR and synthesis.

12.9 WEARABLE TACTILE DISPLAYS

TACTILE interfaces have been used successfully in wearable computing in many applications, ranging from early work on sensory prostheses and navigation aids (Collins, Scadden, and Alden 1977; Bach-y-rita and Kercelz 2003) to recent developments in learning manual skills (Huang et al. 2010) and rehabilitation (Markow et al. 2010; Dimitrijevic, Soroker, and Pollo 1996). Most tactile displays are based on mechanical actuators or electrical stimulation of the skin. Examples of mechanical actuators include vibrators constructed from masses mounted off-center on a spinning motor shaft, solenoids, piezoelectric actuators, and pin arrays. Generally, off-center mass vibrators are used for less precise tasks, such as presenting an alert, due to their longer start and stop times (in the range of 100 ms). Electrical stimulation systems use surface electrodes to create a vibration-like sensation without the use of moving parts. However, depending on the water content of the skin, location, voltage, current, electrical waveform, and electrode size, the perception may range from a tingle, itch, or buzz to a sharp or burning pain (Kaczmarek et al. 1991). While electrical stimulation has many potential benefits, including design simplicity, display resolution, negligible latency, and adjustability of sensation, the changing moisture level of human skin and inconsistent contact during movement makes these systems difficult to use in practice (Lee 2010).

With both electrical and mechanical methods of stimulation, a tactile display interface designer needs to consider where on the body the display will be placed due to the highly varying sensitivity of different areas (Guyton 1991). High-resolution and fidelity displays can be used on the tongue, lips, and fingers (Bach-y-rita and Kercelz 2003), but the wrists (Lee 2010), legs, and back can be surprisingly insensitive. In addition, a tactile display may be masked while a user is in motion due to the self-stimulation caused as clothing moves over the body.

One of the most common uses of a mobile tactile system is to provide better feedback for a graphical user interface (GUI). HCI researchers have shown that tactile feedback can increase users' accuracy when pressing virtual buttons rendered on a touch screen (Hoggan et al. 2008), as is common on many modern mobile phones. Alerts are another common use of mobile tactile displays. A simple example is the "silent," vibration mode on most mobile phones, where the phone vibrates to indicate an incoming call. (Lee and Starner 2010) have shown that a wrist-based, three-vibrator system could be used to communicate richer alert information to the user, such as caller ID. Several projects have explored how best to place actuators on the forearm and determined which features of tactile patterns convey information most efficiently (Lee and Starner 2010; Brown, Brewster, and Purchase 2006; Chen et al. 2008; Borst and Baiyya 2009). The assumption in many of these projects is that tactile alerts may be less distracting than visual or auditory alerts during critical tasks such as driving. This assumption is based on evidence from experimental psychology that users can better divide their attention across modalities as opposed to in the same modality (Wickens and Hollands 1999). Indeed, Lee found that her tactile system is less distracting when the user is performing a visually intensive task than the current practice of retrieving and visually checking the screen of a mobile phone (Lee and Starner 2010). Similarly, soldiers have found tactile alert systems useful to communicate commands covertly and with little distraction while in the field (Gilson, Redden, and Elliot 2007).

Wearable tactile displays have been used for many years as sensory prosthetics or to augment the user's natural sensors.

For example, much work has been performed creating tactile displays that help people who are blind navigate an environment (Collins, Scadden, and Alden 1977). Similarly, directional tactile systems have been used by soldiers for improving situational awareness while clearing buildings (Lindeman et al. 2005). Bach-y-rita's work in the area of sensory prosthetics is of particular interest, as his displays often involve electrical stimulation of the tongue or forehead and have been used for sensory substitution for sight, vestibular balance, and tactile sensation from other parts of the body (Bach-y-rita and Kercelz 2003).

"Passive Haptic Learning" is a recent use of mobile tactile displays. In a series of experiments described by (Huang et al. 2010), subjects learn simple piano melodies while attending other tasks. Participants were equipped with the Mobile Music Touch system; a mobile phone-based music player and a fingerless glove fitted with small vibrators inserted above the thumb and each finger (see Figure 12.22). The melody to be learned is played repeatedly through earphones and, as each note is played, the finger that would be used to play the respective key on the piano (if the user was at a piano and not mobile) is stimulated. Even though participants were required to focus their attention on a distractor task, such as a reading comprehension exam, they still learned the note sequence. A control group who heard the audio repeatedly but did not receive tactile stimulation performed significantly worse. These experiments suggest that passive training for manual tasks might be possible with wearable tactile displays.

Building on the earlier work, Markow et al. (2010) report preliminary results where practice with the Mobile Music Touch system may assist in hand dexterity and sensation rehabilitation in participants with quadriplegia due to partial spinal cord injury. In more mature work, Dimitrijevic et al. (1996) describe a series of experiments using a "Mesh Glove" that uses electrostimulation, sometimes applied below conscious sensation, on the hand to help stroke patients recover arm mobility without active participation. While the Mesh Glove is not designed to be mobile necessarily, the two systems suggest that wearable tactile displays might be worn during the user's everyday life to aid in rehabilitation.

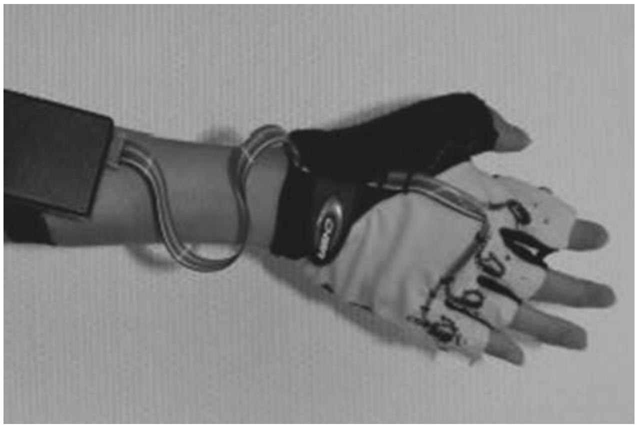


FIGURE 12.22 The Mobile Music Touch glove allows users to learn note sequences while performing other tasks.

12.10 PERFORMANCE EVALUATION

Figure 12.23 illustrates the response time for SR applications running on TIA-P and SR smart module. As SR is using a lightweight operating system (Linux) versus Windows 95 on TIA-P and the SR code is more customized, it has a shorter response time. An efficient mapping of the SR application onto the SR smart module architecture provided a response time very close to real time. To ensure system responsiveness, it was important to provide feedback to the person in near real time.

The lessons learned from tests and demonstrations include: manual intervention process to correct misrecognized words incurs some delay; swapping can diminish the performance of the LT module; the size of display can be as small as a deck of cards.

The required system resources for speech translator software are several times smaller than for the laptop/workstation version, as shown in Table 12.2.

12.11 DUAL PURPOSE SPEECH

In industry, most SR on mobile computers concentrates on the tasks of form filling or simple interface commands and navigation. One reason is that speech interfaces are often socially interruptive when other people are nearby. Speech translation, as with the TIA system above, is a different class of interface. The computer is an essential enabler of the conversation. Lyons et al. (Lyons, Starner, and Plaisted 2004) introduce a different type of conversation enabler in their Dual Purpose Speech work.

Dual Purpose Speech is easiest to discuss using a scenario. Tracy, a wearable user equipped with a head-up display and a Twidder keyboard, is in conversation with a recently introduced colleague. Pressing a button on the keyboard, the

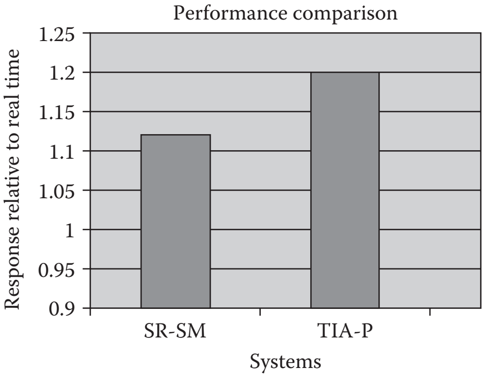


FIGURE 12.23 Response times (lower is better).

TABLE 12.2
Comparison of Required System Resources

	Laptop/ Workstation	Functional Module SR/LT	Optimized Module SR/LT
Memory size	195 MB	53 MB	41 MB
Disk space	1 GB	350 MB	200 MB

wearable user enables SR and says, "Bob, what is your phone number so that I have it for later?"

The wearable recognizes that its user wants to record a phone number and starts the user's contact list application. It attempts to recognize the name spoken and enters that into the application. However, it also saves the speech so that the user can correct the text later if there is an error.

Bob responds "Area code 404."
"404," repeats Tracy.
"555-1212," completes Bob.
"555-1212," continues Tracy who presses another button on her keyboard indicating the interaction is over, "Ok, I have it!"

On Tracy's head-up display a new contact has been made for "Bob (404) 555-1212." When Tracy finishes her conversation, she clicks an "accept" button on the application because she has recognized the information correctly. Tracy could also edit the information or play back the audio recorded during the interaction with Bob. Note that Tracy verbally repeated the information that Bob provided—a good conversational practice. Tracy both confirmed that she understood the information and provided Bob with an opportunity to correct her if necessary. However, this practice is also good from a privacy standpoint. Tracy wears a noise-canceling microphone that is thresholded to record only her own voice and not that of her conversational partners. In this way, Tracy respects the privacy of her colleagues.

Lyons et al. have designed Dual Purpose Speech applications for scheduling appointments, providing reminders for the user, and communicating important information to close colleagues. However, the key point of this research from the perspective of this section is that these applications allow the user to manipulate information on their wearables as part of the process of communicating (thus, the "dual purpose" name). The users may actively format their speech so the system can better understand them, and they may have to correct the system afterwards. However, the interface is manipulated and the information is entered as part of a social process.

This style of interface provides a contrast to the traditional desktop computer where the user's attention is assumed to be dedicated to the interface. Other wearable computing related fields also attempt to create interfaces that are driven by the user's interactions with the environment. For example, Feiner's early augmented reality systems attempted to display appropriate repair instructions based on the user's actions during the repair process (Feiner, MacIntyre, and Seligmann 1993). Such awareness of the user's context and goals may allow wearable computers to be utilized where a user's lack of attentional or physical resources would normally preclude traditional desktop applications.

12.12 PERCEPTION

Just as dexterity is impaired when a user is on-the-go, the user's ability to perceive a wearable's interface is lessened. The vibration and visual interference from a moving background interferes with visual tasks. Background noise and the noise from the body itself affect hearing. The moving of clothes over the body and the coupling of mechanical shock through the body can lessen the user's ability to perceive tactile displays. Sears et al. describe these detriments to mobile interaction caused by environmental and situational factors as "Situationally-Induced Impairments and Disabilities" (Sears et al. 2003). These researchers and others are developing procedures to test human performance in mobile computing tasks in context (in this case, walking a path) (Barnard et al. 2005; Barnard et al. in press). Such research is sorely needed as not enough is known about how to adequately simulate mobile computing scenarios in testing. For example, in Barnard et al.'s work on performing reading comprehension tasks on personal digital assistants while walking, lighting levels affected workload measures more when walking a path than when walking on a treadmill. The community needs to develop understanding about the interactions between mobility, attention, and perception in common mobile computing scenarios in order to adequately develop testing environments for mobile interfaces.

In the past, such work focused on cockpits, both for aviation and automobiles (Wickens and Hollands 1999; Melzer and Moffitt 1997; Velger 1998). However, the U.S. military's Land Warrior project has highlighted the need for such research for dismounted users who are on-the-go (Blackwood 1997). Some researchers have begun exploring mobile output devices for very specific tasks. For example, Krum (Krum 2004) describes experiments with a head-up display that focus on determining how to render overhead views of an area to encourage learning of the layout of the surrounding environment while the user is navigating to a goal on foot. As mobile augmented reality is becoming practical from a technical standpoint, researchers have begun to address perceptual issues. While not a mobile experiment, Laramee and Ware (2002) have investigated head-mounted displays to determine the relative effects of rivalry and visual interference between binocular and monocular displays with varying levels of transparency. As the market determines which mobile contexts are most important for users, experiments such as these will help determine how to design interfaces to least interfere with the user's primary tasks while providing the most value in terms of augmentation.

12.13 RESEARCH DIRECTIONS

The evolution of computing has shown that it takes several years to develop a user interface style that often emerges quite a while after the technology threshold has been passed. The thresholds represent the time when microprocessors have the capability of supporting the indicated form of interface. Figure 12.24 depicts the increase in microprocessor performance (measured in millions of instructions per second, or MIPS) as a function of time. In the early 1960s, Gordon Moore of Intel made the observation/prediction that the capacity of semiconductor chips was doubling every year. Similar trends have been noted for microprocessor speed, magnetic disk storage

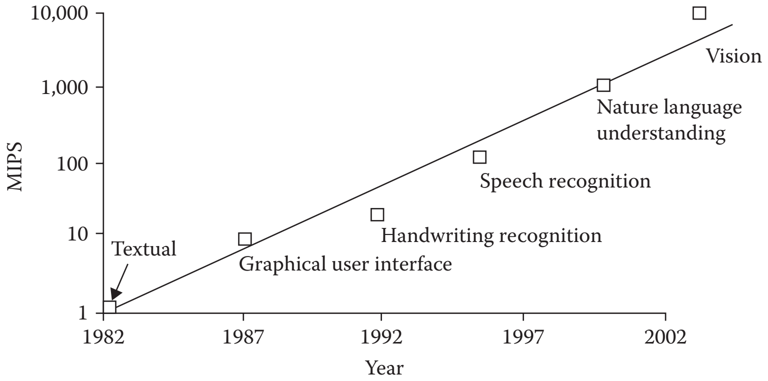


FIGURE 12.24 Minimum performance requirements for various user interaction modalities.

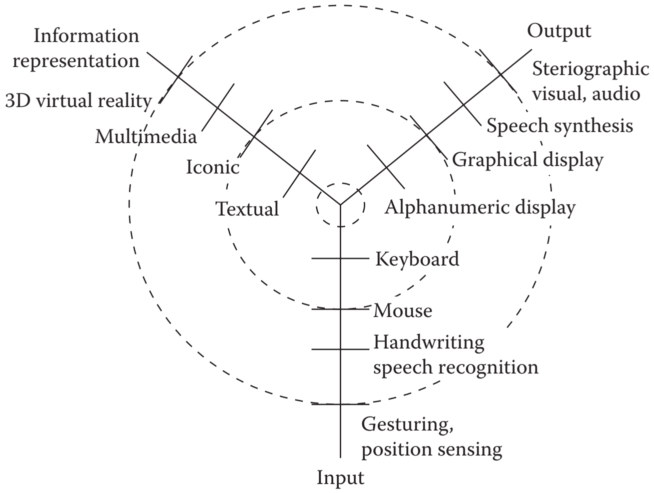


FIGURE 12.25 Kiviat graph for wearable computer use modalities.

capacity, and network bandwidth. The points depicted in Figure 12.24 are the performance thresholds necessary for each of the user interface types. Thus a textual interface requires 1 MIPS, a GUI 10 MIPS, a handwriting interface 30 MIPS, a SR interface 100 MIPS, natural language understanding 1,000 MIPS, and vision understanding, 10,000 MIPS.

Because ease of use is so closely associated with human reaction, it is much more difficult to quantify. There are at least three basic functions related to ease of use: (1) input, (2) output, and (3) information representation.

Figure 12.25 summarizes several points for each of these basic functions. Note that unlike the continuous variables for capacity and performance, the ease of use metrics is discrete.

Just as the performance of microprocessors has increased over time, as shown in Figure 12.24, the characteristics of the user interface shown in Figure 12.25 are also moving out with time. For example, the keyboard with an alpha/numeric display using textual information is representative of timesharing systems of the early 1970s. The keyboard and mouse, graphical output, and iconic desktop are representative of personal computers of the early 1980s. The addition of handwriting recognition input, speech synthesis output, and multimedia information began emerging in the early 1990s. It takes approximately one decade to broadly disseminate new input, output, and informational representations. In the 2000 decade, SR, position sensing, and eye tracking became common inputs. Heads-up projection displays should allow superposition of information onto the user's environment.

12.14 CONTEXT AWARENESS

The next step in the evolution of wearable computers is context awareness. Context-aware computing is aware of a user's state and surroundings and the mobile computer modifies its behavior based on this information. A user's context can be quite rich, consisting of attributes such as physical location, physiological state (such as body temperature, heart rate, and skin resistance), emotional state (such as angry, distraught, or calm), personal history, daily behavioral patterns, and so on. If a human assistant were given such context, he or she would make decisions in a proactive fashion, anticipating user needs. In making these decisions, the assistant would typically not disturb the user at inopportune moments except in an emergency. The goal is to enable mobile computers to play an analogous role, exploiting context information to significantly reduce demands on human attention. Context-aware intelligent agents can deliver relevant information when a user needs that information. These data make possible many exciting new applications, such as augmented reality, context-aware collaboration, wearable assisted living, augmented manufacturing, and maintenance.

12.14.1 Example System: Virtual Coach

The Seating Coach (or Power Wheelchair Virtual Coach) is an intelligent system that can guide power wheelchair users in achieving clinician established goals for body positioning. It was developed at Carnegie Mellon, with the University of Pittsburgh Center for Assistive Technology as the project client. The Seating Coach provides capabilities such as interacting with the user in a manner appropriate to their capability, inferring user capabilities from the data, indicating user compliance, and creating reminders to do past due activities. The Seating Coach sensing and computational infrastructure determines if a user employs auxiliary seating functions according to the physical therapist's prescriptions and coaches him/her to use them in a proper and timely fashion. The Seating Coach can help power wheelchair users to reduce the risk of chronic sores by comparing auxiliary seating function use against a prescription of positions and their durations as established by a physical therapist. It can aid clinicians in tracking results from training and reinforces proper technique to reduce the incidence of injuries caused by improper power wheelchair use.

A power wheelchair allows the user to recline, tilt, elevate the seat, and change leg-rest elevation of the chair. Tilt involves the same change in angle of backrest, seat and leg-rest. Recline changes the backrest angle only and leg-rest elevation changes only the leg-rest angle. The seat elevation only changes the elevation from the ground.

The Seating Coach records sensor data on user position and usage patterns. An array of pressure sensors is distributed over the back rest and seat cushion providing the pressure information to the virtual coach, as shown in Figure 12.26. Three tilt sensors determine the tilt angle of the back rest, seat recline, and leg rest elevation, as illustrated in Figure 12.27. Tilt, recline, and leg rest elevation are monitored for any improper sequences in using seat functions, such as recline without tilt, leg rest elevation without recline, and recline or tilt angles that are too large, as well as any inappropriate use of seat functions during driving. Infrared sensors are used to detect obstacles behind the chair and determine the height of the seat. Pressure sensors are monitored for weight distribution inferring body positions.

The data analysis software extracts underlying user patterns. A clinician-friendly interface allows therapists to prescribe physical activities, rules for proper use of the wheelchair, as well as parameters for user compliance goals. We created a prescription format which is easy to use and expressive enough to cover a range of subjects and conditions. The prescription encompasses all the power seat functions (PSFs) and sets limits for the user. The prescription is specified using the following information: activity type, parameter, duration of the activity, and time gap after which to repeat the activity, and alert after specified number of rule violations (Table 12.3).

After entering a usage prescription, the clinician can periodically monitor the wheelchair user's compliance to those recommendations. Reminders are generated to prompt the user to comply while alerts indicate noncompliance and are sent to the user, as shown in Figure 12.28.

In addition, a Wizard of Oz experiment was conducted where users made selections from a variety of feedback modalities and preferences to create a user interface (Liu et al. 2010). Nine PSF users and six clinicians were recruited for this study. The subjects reviewed modalities with various properties using a computer demonstration program with supplemental devices. Their preferences and suggestions were collected using a questionnaire and interviews. An animation of PSF usage tasks was preferred because it conveyed

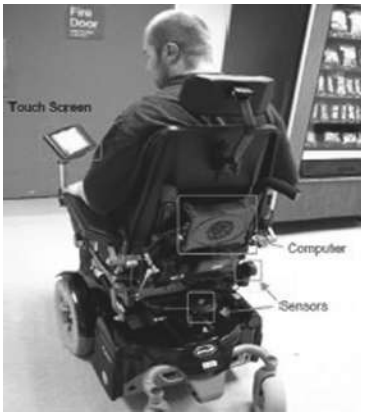


FIGURE 12.26 Power Wheelchair Virtual coach.

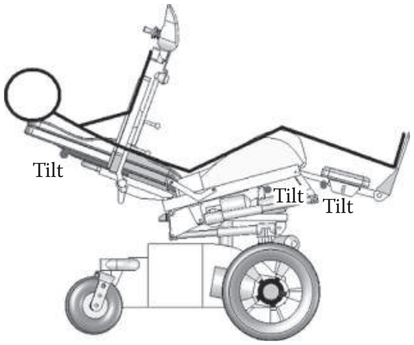


FIGURE 12.27 Tilt function and placement of sensors.

TABLE 12.3
Sample Prescription, Filled by the Clinician

Activity	Parameter			Duration			Gap			Alert after
	Min	Ideal	Max	Min	Ideal	Max	Min	Ideal	Max	
Tilt	25°	30°	35°	25 sec	30 sec	35 sec	20 min	30 Min	2 hrs	10
Recline	10°	15°	20°	4 mins	5 mins	6 mins	4 hrs	5 hrs	6 hrs	15
Feet Elevation	25°	30°	35°	50 sec	1 min	1 min 10 sec	1 hr 30 mins	2 hrs	2 hrs 30 mins	20
Pressure	0	60 mm	200 mm	0 sec	0 sec	30 mins	0 sec	0 sec	0 sec	5

General Tilt angle: Min 10, Ideal 20, Max 30
General Recline angle: Min 10, Ideal 30, Max 40

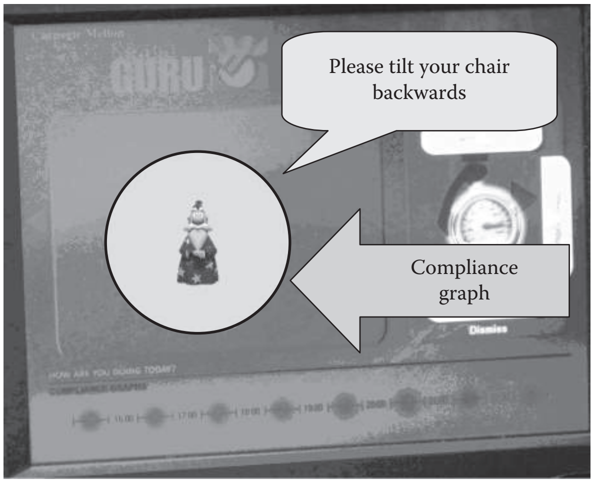


FIGURE 12.28 Example system: virtual coach.

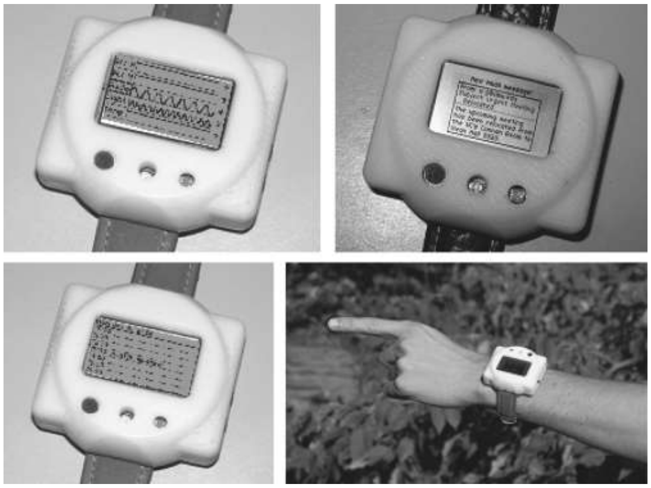


FIGURE 12.29 eWatch—a proactive assistant.

essential information. The subjects rank ordered the interaction options as a function of situation. For example, 40% of the subjects selected vibration for the reminding theme in the noisy restaurant scenario, and 46.7% selected speech for the reminding theme in the home scenario. As another example, ranking of vibration location on the seat had armrest ranked highest (60%), and headrest as lowest (6.7%). These studies will inform user interaction designs for virtual coaches.

12.14.2 Example: eWatch, a Proactive Assistant

Our research on context-aware computing employs unsupervised machine learning techniques to combine real-time data from multiple sensors into a model of behavior that is individualized to the user. The eWatch is a wearable sensing, notification, and computing platform built into a wrist watch form factor making it highly available, instantly viewable, ideally located for sensors, and unobtrusive to users (Smailagic and Siewiorek 2005). Bluetooth communication provides a wireless link to a cellular phone or stationary computer. eWatch senses light, motion, audio, and temperature and provides visual, audio, and tactile notification. The system provides ample processing capabilities with multiple day battery life enabling realistic user studies. Figure 12.29 shows a few representative eWatch screenshots: sensor waveforms, calendar, and messages. Figure 12.30 illustrates sensors and main hardware components on eWatch. We developed a wearable computing platform with power-aware hardware and software architectures, and showed how online nearest neighbor classification can identify and recognize a set of frequently visited locations.

Knowing about the user's location is an important aspect of a context-aware system. Using eWatch we developed a system that identifies previously visited locations. Our method uses information from the audio and light sensor to learn and distinguish different environments. We recorded and analyzed the audio environment and the light conditions at several different locations. Experiments showed that locations have unique background noises such as car traffic, talking, noise of computers, air conditioning and television. The light sensor sampling at a high frequency can also provide additional information beyond the brightness of the location. We observed that the frequency characteristics of light conditions tend to remain constant in most locations. For our study, audio data was recorded with the built-in microphone at a sample rate of 8 kHz and the light sensor at a frequency of 2048 Hz. At every location five consecutive recordings of audio and light were taken, separated by 10-second pauses. For every recording we sampled the microphone for 4 seconds (32,000 samples) and the light sensor for 0.5 seconds (1024 samples). The recorded data was then compressed and stored into flash memory. Locations frequently visited by the user were recorded; the rooms of the user's apartment (living room, kitchen, bedroom, bathroom), their office, the lab, different street locations on the way to university, the interior of a bus, and several restaurants and supermarket. Each location was visited multiple times on different days. In total, we collected 600 recordings at 18 different locations.

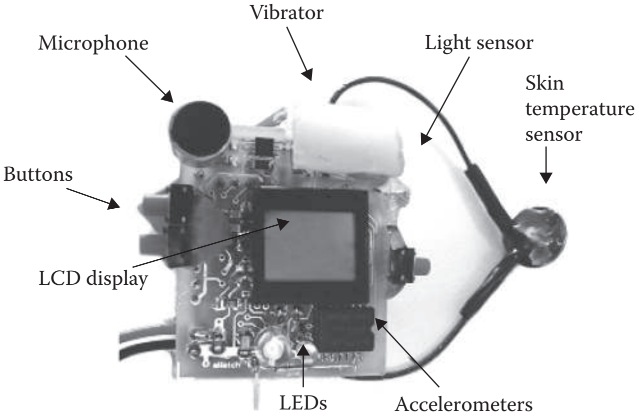


FIGURE 12.30 The eWatch hardware.

We estimated the power spectral density of the recorded sensor data using Welch's method. A 128-point fast Fourier transform was calculated for a sliding window over the complete recording and averaged over frequency domain coefficients for all windows.

The result is a smoothed estimation of the power spectral density. To reduce the number of feature components, the Principal Component Analysis was used. The dimensionality of the feature vector was reduced to its first five principal components. To visualize the feature space, Figure 12.31 shows the first three components of the feature vectors after a Linear Discriminant Analysis transformation.

The nearest neighbor method with a five-fold cross-validation was used for classification. Three different feature sets were evaluated: features from the light sensor only, microphone only, and both sensors combined. As expected, the combination of both sensors gave the best results in identifying the location. The classification with the light sensor alone gave an overall result of 84.9% correctly classified samples. The classifier confused the *lab* and *office* location and also the *bus* with the *street*. This occurred because both location pairs can have similar light conditions. Using only the audio sensor the overall recognition accuracy was 87.4%. The *office* and *apartment* location were confused in this case. Both sensors combined gave the best result of 91.4%. Locations that could not be distinguished well with only one sensor were classified more accurately with both sensors combined.

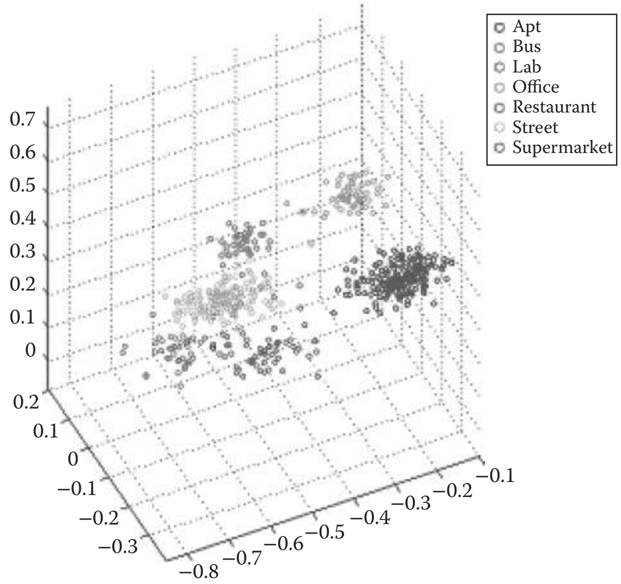


FIGURE 12.31 Learning context: audio and light sensor clustering.

12.15 WEARABLE COGNITIVE AUGMENTATION

An important goal of our research is to determine the cognitive state of a user—especially the user's cognitive load—from external observations. Knowing the user's cognitive state would enable development of proactive cognitive assistants that anticipate user needs much like a human assistant does.

What makes this attempt possible is an unprecedented advance in measuring and understanding brain activity during complex cognition using functional magnetic resonance imaging (fMRI) (Figure 12.32). The brain activity measured with fMRI is only one step removed from the neural activity itself. fMRI provides a measure of the oxygenated hemoglobin in the capillary beds in which the neural activity is occurring. Routinely used protocols in neurocognitive research on advanced MRI scanners sample the entire cortex approximately once per second. It is feasible to pursue a research plan to recognize some of the brain/cognitive states that should be amenable to improvement, and then develop an intelligent tutoring system that uses the fMRI-measured brain activation to guide the tutoring, to infer current mental states, and to rapidly guide the learner to desired mental states.

There is also a maximum on the total activation across cortical areas. Such a system-wide capacity constraint might be expected to operate when subjects co-perform two tasks that draw on nonoverlapping brain areas. The requirement of nonoverlap assures that any constraint on performance is not due just to competition for the same neural mechanisms.

In a study that found evidence for such a constraint, the two tasks were auditory sentence comprehension and mental rotation (Just et al. 2001). If there were no system-wide

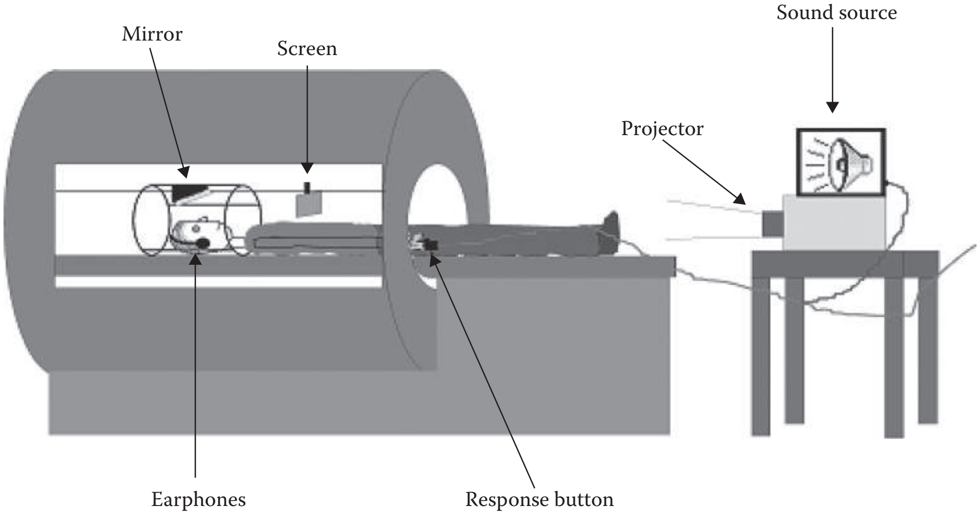


FIGURE 12.32 Functional magnetic resonance imaging experiment configuration.

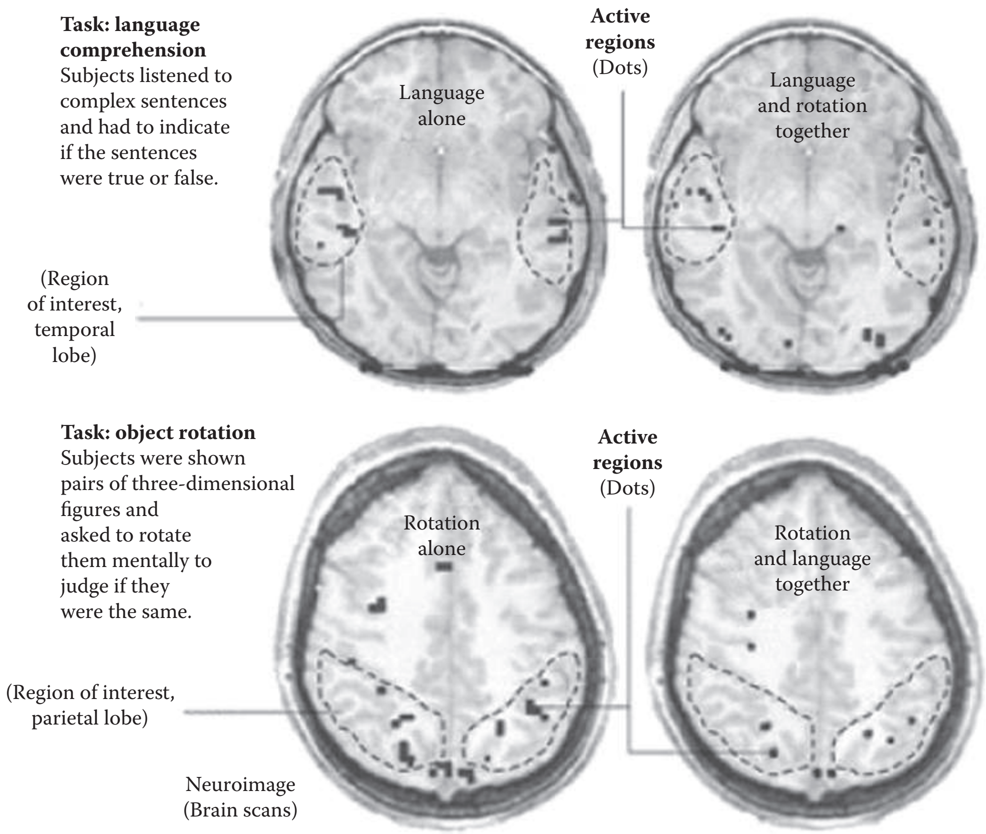

Task: language comprehension
Subjects listened to complex sentences and had to indicate if the sentences were true or false.
Language alone
Active regions (Dots)
Language and rotation together
(Region of interest, temporal lobe)
Task: object rotation
Subjects were shown pairs of three-dimensional figures and asked to rotate them mentally to judge if they were the same.
Rotation alone
Active regions (Dots)
Rotation and language together
(Region of interest, parietal lobe)
Neuroimage (Brain scans)

FIGURE 12.33 Study of cortex.

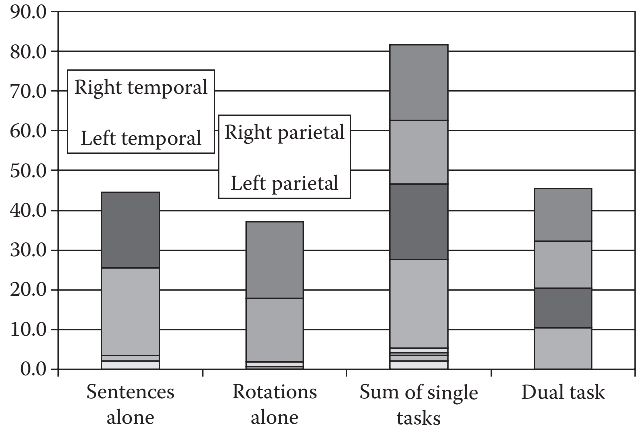


FIGURE 12.34 Activation volume in dual task comparing to single task.

capacity constraint, one would expect that because the two tasks draw on different neural substrates (language and spatial-related areas respectively), the activation in the dual task would simply be the union of the activations in each of the two single tasks. However, the activation in the dual task was far less than the union of the two single tasks. The activation associated with each individual task decreased by 30%–50% in the dual task condition, as shown in the representative brain slices of individual subjects (Figure 12.33), and in the graph presenting the group data results (Figure 12.34). The decrease in the dual task condition applied to 17 of the 18 subjects. Thus there appears to be a detectable upperbound on the total amount of activation that can be sustained in a set of cortical areas. (This study was widely applied to the question of what happens in the brain during driving and using a cell phone simultaneously.)

12.16 CONCLUSION AND FUTURE CHALLENGES

Wearable computers are an attractive way to deliver a ubiquitous computing system’s interface to a user, especially in non-office-building environments. The biggest challenges in this area deal with fitting the computer to the human in terms of interface, cognitive model, contextual awareness, and adaptation to tasks being performed. These challenges include the following:

- User interface models: What is the appropriate set of metaphors for providing mobile access to information (i.e., what is the next “desktop” or “spreadsheet”)? These metaphors typically take over a decade to develop (i.e., the desktop metaphor started in early 1970s at Xerox PARC (Palo Alto Research Center Incorporated) and required over a decade before it was widely available to consumers). Extensive experimentation working with end-user applications will be required. Furthermore, there may be a set of metaphors each tailored to a specific application or a specific information type.

- Input/output modalities: While several modalities mimicking the input/output capabilities of the human brain have been the subject of computer science research for decades, the accuracy and ease of use (i.e., many current modalities require extensive training periods) are not yet acceptable. Inaccuracies produce user frustrations. In addition, most of these modalities require extensive computing resources which will not be available in low-weight, low-energy wearable computers. There is room for new, easy-to-use input devices such as the dial developed at Carnegie Mellon University for list-oriented applications.
- Quick interface evaluation methodology: Current approaches to evaluate a human computer interface requires elaborate procedures and with scores of subjects. Such an evaluation may take months and is not appropriate for use during interface design. These evaluation techniques should especially focus on decreasing human errors and frustration.
- Matched capability with applications: The current thought is that technology should provide the highest performance capability. However, this capability is often unnecessary to complete an application and enhancements such as full-color graphics require substantial resources and may actually decrease ease of use by generating information overload for the user. Interface design and evaluation should focus on the most effective means for information access and resist the temptation to provide extra capabilities simply because they are available.
- Context-aware applications: How do we develop social and cognitive models of applications? How do we integrate input from multiple sensors and map them into user social and cognitive states? How do we anticipate user needs? How do we interact with the user? These, plus many other questions, have to be addressed before context-aware computing becomes possible. Some initial results have been reported in (Krause, Smailagic, and Siewiorek 2006).

REFERENCES

Anhalt, J., A. Smailagic, D. Siewiorek et al. 2001. Towards context aware computing. *IEEE Intell Syst* 6(3):38–46.

Azuma, R. 1997. A survey of augmented reality. *Presence* 6(4):355–86.

Bach-y-Rita, P., and S. Kercelz. 2003. Sensory substitution and the human-machine interface. *Trends Cogn Sci* 7(12):541–6.

Barnard, L., J. S. Yi, J. A. Jacko, and A. Sears. 2005. An empirical comparison of use-in motion evaluation scenarios for mobile computing devices. *Int J Hum Comput Stud* 62:487–520.

Barnard, L., J. Yi, J. A. Jacko, and A. Sears. In press. Capturing the effects of context on human performance in mobile computing systems. To appear: *Personal and Ubiquitous Computing*.

Blackwood, W. 1997. *Tactical Display for Soldiers*. Washington, DC: National Academy of Sciences.

Bodine, K., and F. Gemperle. 2003. Effects of functionality on perceived comfort of wearables. In *Proceedings of Seventh IEEE International Symposium on Wearable Computers*, 57–60. Los Alamitos, CA: IEEE Computer Society Press.

Borst, C., and V. Baiyya. 2009. A 2d haptic glyph method for tactile arrays: Design and evaluation. In *Proceedings of the World Haptics Conference* 599–604. IEEE Computer Society Press.

Brown, L., S. Brewster, and H. Purchase. 2006. Multidimensional tactons for non-visual information presentation in mobile devices. In *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services*, 231–238. Helsinki, Finland: ACM.

Chen, H. -Y., J. Santos, M. Graves, K. Kim, and H. Tan. 2008 Tactor localization at the wrist. In *Proceedings of EuroHaptics*, 209–218. Madrid, Spain.

Christakos, C. K. 1998. *Optimizing a Language Translation Application for Mobile Use*. Master's thesis, Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA.

Clawson, J., K. Lyons, T. Starner, and E. Clarkson. 2005 The impacts of limited visual feedback on mobile text entry using the mini-QWERTY and Twiddler keyboards. In *Proceedings of IEEE International Symposium on Wearable Computers* 170–177. Osaka, Japan: IEEE Computer Society Press.

Collins, C., L. Scadden, and A. Alden. 1977. Mobile studies with a tactile imaging device. In *Proceedings of the Fourth Conference on Systems & Devices For The Disabled*, Seattle, WA.

Dimitrijevic, M., N. Soroker, and F. Pollo. 1996. Mesh glove electrical stimulation. *Sci Med* 3:54–63.

Feiner, S., B. MacIntyre, and D. Seligmann. 1993. Knowledge-based augmented reality. *Commun ACM* 36(7):52–62.

Frederking, R. E., and R. Brown. 1996. The Pangloss-lite machine translation system: Expanding MT horizons. In *Proceedings of the Second Conference of the Association for Machine Translation in the Americas*, 268–72. Montreal, Canada: AMTM Press.

Gemperle, F., C. Kasabach, J. Stivoric, B. Bauer, and R. Martin. 1998. Design for wearability. In *Second International Symposium on Wearable Computers*, 116–22. Pittsburgh, PA: IEEE Computer Society Press.

Gilson, R., E. Redden, and L. Elliot, eds. 2007. *Remote Tactile Displays for Future Soldiers*. Army Research Laboratory Technical Report ASL-SR-0152.

Guyton, A. 1991. *Basic Neuroscience and Anatomy*. Philadelphia, PA: W.B. Saunders Co.

Hoggan, E., S. Brewster, and J. Johnston. 2008. Investigating the effectiveness of tactile feedback for mobile touchscreens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1573–1582. Florence, Italy: ACM Press.

Huang, K., D. Kohlsdorf, T. Starner, C. Ahlrichs, L. Ruediger, E. Do, and G. Weinberg. 2010. Mobile music touch: Mobile tactile stimulation for passive learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 791–800. Atlanta, GA: ACM Press.

Just, M., Carpenter, P., Keller, T., Emery, L., Zajac, H., Thulborn, K. 2001. Interdependence of nonoverlapping cortical systems in dual cognitive tasks. *NeruoImage,* 14, 417–426.

Kaczmarek, K., J. Webster, P. Bach-Y-Rita, and W. Tompkins. 1991. Electrotactile and vibrotactile displays for sensory substitution systems. *IEEE Trans Biomed Eng* 38:1–16.

Kollmorgen, G. S., D. Schmorrow, A. Kruse, and J. Patrey. 2005. The cognitive cockpit-state of the art human-system integration. In *Proceedings of 2005 Interservice/Interindustry Training Simulation and Education Conference*, Arlington, VA: DARPA.

Krause, A., A. Smailagic, and D. P. Siewiorek. 2006. Context-aware mobile computing: Learning context-dependent personal preferences from a wearable sensor array. *IEEE Trans Mob Comput* 5(2):113–27.

Krum, D. 2004. Wearable computers and spatial cognition. Ph.D Thesis, Georgia Institute of Technology, Atlanta, GA.

Laramee, R., and C. Ware. 2002. Rivalry and interference with a head mounted display. *ACM Trans Comput Hum Interface* 9(3):238–51.

Lee, S. 2010. Buzzwear: Supporting multitasking with wearable tactile displays on the wrist. Unpublished doctoral diss., Georgia Institute of Technology, School of Interative Computing, Atlanta, GA.

Lee, S., and T. Starner. 2010. BuzzWear: Alert perception in wearable tactile displays on the wrist. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 433–42. Atlanta, GA: ACM Press.

Li, K. F., H. W. Hon, M. J. Hwang, and R. Reddy. 1989. The Sphinx speech recognition system. In *Proceedings of the IEEE ICASSP*, 170–177. Los Alamitos, CA: IEEE Computer Society Press.

Lindeman, R., J. Sibert, E. Mendez-Mendez, S. Patil, and D. Phifer. 2005. Effectiveness of directional vibrotactile cuing on a building-clearing task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 271–80. Portland, Oregon: ACM Press.

Liu, H. -Y., R. Cooper, R. Cooper, A. Smailagic, D. Siewiorek, D. Ding, and F. C. Chuang. 2010. Seating virtual coach: A smart reminder for power seat function usage. *Technol Disabil* 21(4):41–8.

Lyons, K., C. Skeels, T. Starner, B. Snoeck, B. Wong, and D. Ashbrook. 2004. Augmenting conversations using dual-purpose speech. In *Proceedings of User Interface and Software Technology*, 237–46. Montreal, Canada.

Lyons, K., and T. Starner. 2001. Mobile capture for wearable computer usability testing. In *Proceedings of International Symposium on Wearable Computers*, 170–177. Zurich, Switzerland: IEEE Computer Society Press.

Lyons, K., T. Starner, and D. Plaisted. 2004. Expert typing using the twiddler one handed chord keyboard. In *Proceedings of the IEEE International Symposium on Wearable Computers*, 94–101. Montreal, Canada.

Markow, T., K. Ramakrishnan, K. Huang, T. Starner, S. Eicholts, Garrett, H. Profita, A. Scarlata, C. Schooler, A. Tarun, D. Backus. 2010. Mobile Music Touch: Vibration Stimulus as a Possible Hand Rehabilitation Method. In *Proceedings of Pervasive Computing Technologies for Healthcare*, Munich, Germany.

Martin, T. 1999. *Balancing Batteries, Power, and Performance: System Issues in CPU Speed-Setting for Mobile Computing*. PhD Thesis, Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA, USA.

Melzer, J., and K. Moffitt. 1997. *Head mounted displays: Designing for the user*. New York: McGraw-Hill.

Ockerman, J. 2000. *Task Guidance and Procedure Context: Aiding Workers in Appropriate Procedure Following*. Technical Report, Georgia Institute of Technology, Atlanta, GA.

Oulasvirta, A., S. Tamminen, V. Roto, and J. Kuorelahit. 2005. Interaction in 4-second bursts: The fragmented nature of attentional resources in mobile HCI. In *Proceedings of SIGCHI Conference on Human Factors in Computing Systems*, 919–28. ACM Press.

Ravishankar, M. 1996. *Efficient Algorithms for Speech Recognition*. PhD thesis, Carnegie Mellon University, School of Computer Science, Pittsburgh, PA, USA.

Reilly, D. 1998. *Power Consumption and Performance of a Wearable Computing System*. Masters thesis, Carnegie Mellon University, Electrical and Computer Engineering Department, Pittsburgh, PA, USA.

Sears, A., M. Lin, J. Jacko, and Y. Xiao. 2003. When computers fade: Pervasive computing and situationally-induced impairments and disabilities. In *Proceedings of HCII 2003*, 1298–302. Amsterdam, The Netherlands: Elsevier.

Siewiorek, D. P. 2002. Issues and challenges in ubiquitous computing: New frontiers of application design. *Commun ACM* 45(12):79–82.

Siewiorek, D. P, A. Smailagic, L. Bass, J. Siegel, R. Martin, and B. Bennington. 1998. Adtranz: A mobile computing system for maintenance and collaboration. In *Proceedings of the 2nd IEEE International Conference on Wearable Computers*, 25–32. Pittsburgh, PA: IEEE Computer Society Press.

Siewiorek, D. P., A. Smailagic, and J. C. Lee. 1994. An interdisciplinary concurrent design methodology as applied to the Navigator wearable computer system. *J Comput Softw Eng* 2(2):259–92.

Smailagic, A. 1997. ISAAC: A voice activated speech response system for wearable computers. In *Proceedings of the IEEE International Conference on Wearable Computers*, 183–184. Cambridge, MA: IEEE Computer Society Press.

Smailagic, A., and D. Siewiorek. 1993. A case study in embedded system design: The VuMan 2 wearable computer. *IEEE Des Test Comput* 10(3):56–67.

Smailagic, A., and D. P. Siewiorek. 1994. The CMU mobile computers: A new generation of computer systems. In *Proceedings of the IEEE COMPCON 94*, 467–73. Los Alamitos, CA: IEEE Computer Society Press.

Smailagic, A., and D. Siewiorek. 1996. Modalities of interaction with CMU wearable computers. *IEEE Pers Commun* 3(1):14–25.

Smailagic, A., D. P. Siewiorek, R. Martin, and J. Stivoric. 1998. Very rapid prototyping of wearable computers: A case study of VuMan 3 custom versus off-the-shelf design methodologies. *J Des Autom Embed Syst* 3(2–3):219–32.

Smailagic, A., D. P. Siewiorek, and D. Reilly. 2001. CMU wearable computers for real-time speech translation. *IEEE Pers Commun* 8(2):6–12.

Smailagic, A., D. P. Siewiorek, Maurer, U., Rowe, A., Tang, K. 2005. eWatch: Context Sensitive System Design Case Study, Proceedings of IEEE Conference of VLSI, IEEE Computer Society Press, Orlando, FL.

Starner, T. 2001. The challenges of wearable computing: Part 1+2. *IEEE Micro* 21(4):44–67.

Starner, T., C. Snoeck, B. Wong, and R. McGuire. 2004. Use of mobile appointment scheduling devices. In *Proceedings of ACM Conference Human Factors in Computing Systems*, 1501–4. Vienna, Austria: ACM Press.

Stein, R., S. Ferrero, M. Hetfield, A. Quinn, and M. Krichever. 1998. Development of a commercially successful wearable data collection system. In *Proceedings of IEEE International Symposium on Wearable Computers*, 18–24. Pittsburgh, PA: IEEE Computer Society Press.

Velger, M. 1998. *Helmet-Mounted Displays and Sights*. Norwood, MA: Artech House, Inc.

Wickens, C. D., and J. Hollands. 1999. *Engineering Psychology and Human Performance*. 3rd ed. Englewood Cliffs, NJ: Prentice Hall.

13 Design of Fixed, Portable, and Mobile Information Devices

Michael J. Smith and Pascale Carayon

CONTENTS

13.1 INTRODUCTION

In this second revision of our chapter, we discuss fixed computer workstations and mobile use of information technology (IT), which we have termed "portable information devices" (PIDs). These portable technologies are now in use in almost every venue and human activity, and the nature of their characteristics and activities of use do not lend them to traditional fixed workstation considerations. This introduces a host of potential ergonomic concerns related to the design of work areas (and activities) where PIDs and other forms of computing are used. There have been decades of research and applications that have defined important considerations in the ergonomic design of fixed computer work areas (Grandjean 1987; ANSI/HFES 100-1988; Smith and Cohen 1997; Smith, Carayon, and Cohen 2003, 2008; BSR/HFES-100 2005; ANSI/HFES-100 2007; OSHA 2008; WorkSafeBC 2009). However, much less has been done to define the design of work areas for PIDs and mobile computing. In this chapter, we propose some ideas and considerations for dealing with ergonomic concerns for these mobile technologies in addition to updating information on fixed computer workstation applications.

Ergonomics is the science of fitting the environment and activities to capabilities, dimensions, and needs of people to enhance their performance, safety, and health. Ergonomic knowledge and principles are applied to adapt activities and environmental conditions to the physical, psychological, and social nature of the person and groups. Computer workstation design is more than just making the computer interfaces easier to use, or making furniture adjustable in various dimensions. It also involves integrating design considerations with the work environment, task requirements, and psychosocial aspects of work and job design. Critical considerations for good ergonomic practice are as follows:

- To reduce biomechanical loading on the back and joints to the lowest practical forces
- To keep repetition of body parts as low as practical
- To keep the back, neck, and joints in good postures
- To reduce the amount of time of static postures
- To reduce the duration and extent of highly repetitive motions or high forces
- To resign environmental conditions so that people can easily see and hear
- To provide frequent rest breaks from activities for resting and recovery
- To provide healthy psychosocial working conditions

A fundamental perspective discussed in this chapter is that the work area (workstation) design influences employees' comfort, health, motivation, and performance. We will examine basic ergonomic considerations (principles, practices, concerns) that can be used to develop guidance for the design of work areas (workstations) for the use of computing and related IT products such as PIDs. Some of today's technologies have the capability to directly interact with one another, sometimes without human intervention, whereas older technologies require human action. The wide myriad of computer technologies, environments of use, interaction schemes, and activities of use make specific guidance for PIDs complex and difficult. However, we will provide general guidance that has the possibility to reduce musculoskeletal stress and to enhance employees' physical comfort.

The relationship between the user and technology is a system in which the constraints of one component can affect the performance of the other. Various aspects of the work system such as the task requirements, the work demands, the environment, and the workstation influence how effectively and comfortably the technology can be used by the user (Smith and Sainfort 1989; Smith and Carayon 1995; Carayon and Smith 2000). One of the consequences of this is that the design of the workspace limits the nature and effectiveness of the interaction between the user and the technology. For instance, inadequate space for carrying out physical activities can lead to constrained postures, which together with long-lasting static loading may produce discomfort in the muscles and joints. This leads to symptoms of tiredness, muscle aches, and muscle and joint pain. In addition, heavy workload, chronic high repetition, and other biomechanical strains can cause similar problems. The adverse postural, repetition, and workload exposures can lead to reduced performance and productivity, and in the long run, they may also affect employees' well-being and health.

Today, people carry their computing and communications on their person and engage in activities in many diverse work areas without a workstation. There is a potential for serious ergonomics risks from the new modes of using new technologies.

13.2 HISTORICAL AND RECENT ERGONOMIC PERSPECTIVES ON COMPUTER WORKSTATIONS

Forty-five years ago, a person would interact with several different types of information sources and technologies when engaging in activities. Computer terminals that were connected to mainframe computers were just coming into common use in government, insurance, and banking industries. An employee might look at hard-copy documents, take notes with a pen and a paper, use a fixed location telephone, talk face-to-face with colleagues, type on a typewriter, and use a fixed computer terminal. The diversity of activities led people to actively move around during the day. Then, 35 years ago, many employees started to spend most of their workday in sedentary work, that is, sitting in front of a computer terminal. This type of human–machine system led to restricted physical movement, and much greater mental attention of the employees was directed toward the computer monitor. Over time, the terminals that were connected to mainframe computers were replaced by PCs connected to a network of computers. Later, laptop computers and easily portable IT devices were increasingly used.

Today, millions of employees still work primarily at fixed workstations on PCs. But there has also been a substantial shift to the use of laptop computers and PIDs, which seems to be the trend for the future. Today, we have a variety of circumstances in multidimensional environments where people operate from fixed workstations part of the time, laptops part of the time, and mobile connections to the Internet at other times. In many instances, people are again on the move and not completely in fixed, sedentary situations. Although this moving around is better for the muscles and joints than constant sedentary sitting, it has introduced new ergonomics challenges. The interaction with technology while moving can create awkward postures and loads that can produce strain on the body. Hence, we have a whole new set of ergonomic concerns and benefits due to the dynamics that new technologies have afforded.

Since the initial work looking at health issues of computerized work by Hultgren and Knave (1973), Ostberg (1975), Gunnarsson and Ostberg (1977), Cakir, Hart, and Stewart (1979), Grandjean (1979, 1980), and Smith et al. (1980, 1981), there have been thousands of research studies from every corner of the globe examining the working conditions of computer users and their health and productivity issues. There have been several international conferences devoted to these issues starting in 1980, and these conferences have continued on a regular basis to the present. Suffice it to say that interest in human factors and ergonomics, usability and human–computer interaction issues of new IT will be of interest for decades to come.

The findings from research on health of computer users have generally indicated that poor ergonomic conditions are associated with visual discomfort, musculoskeletal discomfort and pain, and psychological distress. Research indicates that long-term adverse ergonomic conditions lead to chronic employee aches and pains in the upper extremities and back, and may involve not only muscles but also other soft tissues, such as tendons and nerves. Across many types of jobs, long-lasting, adverse ergonomic conditions have been shown to lead to a deterioration of joints, ligaments, tendons, and nerves (Hagberg et al. 1995; Bernard 1997; Carayon, Smith, and Haims 1999). Reviews of field studies for fixed computer workstations (Grandjean 1979, 1980, 1987; NAS 1983; Bergqvist 1984; Smith 1984, 1987, 1997) have shown that poor ergonomics and working conditions for computer users are related to visual, musculoskeletal, and psychological discomfort and health disorders.

Findings from recent research on computer users are similar to that from research in the 1970s and 1980s that indicated employees who work extensively with computers experience upper extremity musculoskeletal discomfort and health complaints.

Robertson, Huang, and Larson (2009) conducted a survey of 1259 computer users at a large manufacturing company in the United States. In their study, 48% reported eye strain, 45% headaches, 43% neck pain, 40% shoulder pain, 36% wrist pain, and 35% low back pain. Significant associations were found between all visual and body part discomfort symptoms and the number of hours working at computers. About one-half of the computer users made appropriate changes to their workstations or work habits to address their visual and musculoskeletal discomfort. Computer users who made positive changes reported less discomfort.

Eltayeb et al. (2007, 2009) conducted a prospective cohort study over a 2-year period examining upper extremity health complaints among 264 computer users at the Dutch national social security office. They examined workstation considerations, employees' work postures, job demands, and psychosocial work characteristics. The first year prevalence of musculoskeletal symptoms was 33% for the neck, 31% for the shoulders, 11% for the hand, 12% for the upper arms, 6% for the elbows, and 7% for the lower arms/wrist. The results were consistent over the 2-year period, but there was an increase in arm/hand prevalence to 21% by the end of the second year. Four main predictors were identified for neck and shoulder complaints: (1) irregular head and body postures, (2) task difficulty (job demands), (3) number of hours per day working on the computer, and (4) a history of musculoskeletal complaints. Two main predictors were identified for forearm/hand complaints: (1) job time pressures (job demands) and (2) a history of musculoskeletal complaints.

Eltayeb et al. (2008) conducted a second study of computer users in a mobile telecommunications company and three banks in Sudan to compare the prevalence of musculoskeletal complaints with those they found in the Netherlands. They surveyed 282 employees and received 250 responses. The 1-year prevalence for musculoskeletal symptoms of the neck was 64% and for the shoulder 41%.

Gerr et al. (2002) conducted a prospective study of computer users who were newly hired for jobs requiring at least 15 hours of computer use per week. They followed 642 of these computer users for 3 years. They evaluated workstation characteristics and medical and psychosocial risk factors. The 1-year incidence for neck/shoulder musculoskeletal symptoms was 58%, and the incidence of neck/shoulder musculoskeletal disorders was 35%. The 1-year incidence for hand/arm musculoskeletal symptoms was 39%, and the incidence of hand/arm musculoskeletal disorders was 21%. Overall, more than 50% of computer users reported musculoskeletal symptoms during the first year after starting the job. Predictors for neck/shoulder problems were gender, age, ethnicity, and prior history of neck/shoulder pain. Predictors for hand/arm musculoskeletal problems were gender, prior history of hand/arm pain, prior computer use, and having children at home. The postures of the elbows, the head tilt, and the presence of arms rests influenced the prevalence of neck/shoulder problems, whereas wrist radial deviation when using a mouse, keyboard location and height, and key activation force influenced the prevalence of hand/arm problems. The number of hours of keying per week was associated with a greater risk of hand/arm musculoskeletal symptoms or disorders (Marcus et al. 2002).

13.3 DESIGN OF FIXED WORKSTATIONS

Workstation design is a major element in ergonomic strategies for improving user comfort and particularly for reducing musculoskeletal problems. Often the task requirements will have a role in defining the layout and dimensional characteristics of the workstation. The relative importance of the screen, input devices, and hard copy (e.g., source documents) depends primarily on the task, and this can influence the design considerations necessary to improve operators' performance, comfort, and health. Many studies have examined fixed computer workstation design issues and generally have found that the quality of the fit between the user and the interactive devices being used has a substantial role in the user's musculoskeletal comfort and musculoskeletal pain (Smith, Carayon, and Cohen 2003, 2008). For instance, Cohen et al. (1995) identified the following working conditions that influenced work postures and led to undue loads on the musculoskeletal system:

- Static postures of the trunk, neck, and arms
- Awkward twisting and reaching motions
- Poor lighting and glare
- Placement of the keyboard on uneven working surfaces
- Insufficient work surface space
- Insufficient knee and toe space
- The inability for the chair armrests to fit under the working surfaces
- Chairs with poor back and shoulder support, inadequate padding in the backrest and seat pan, arm rests that did not fit under working surfaces, and a lack of appropriate seat pan height adjustment

Derjani-Bayeh and Smith (1999) and Smith and Derjani-Bayeh (2003) conducted a prospective intervention study to examine the benefits of ergonomic redesign for computer users at fixed workstations. The study took place in a consumer products call center where shoppers could order products from a catalog using a telephone or an Internet site. There were three ergonomic interventions studied. In the first condition, ergonomics experts modified current workstation configurations to maximize their fit with the incumbent. In the second condition, new workstation accessories (keyboard tray, monitor holder, document holder, wrist rest, foot rest, and/or task lighting) were added as needed to improve the employees' fit with their workstations and general environment. In the third condition, the same factors in the second condition were added but in addition a new chair with multiple adjustments was also added.

A total of 80 volunteer subjects participated in the study. They were drawn from a larger pool of volunteers. The participants for the third condition were randomly selected from the larger pool, and then subjects for conditions 1 and 2 were matched to these selections based on the type of job, age, gender, and length of experience with the company. Baseline measurements of self-reported health status were collected using a questionnaire survey. Follow-up measurements were taken directly after implementation of the ergonomic improvements and then 12 months later. In addition, productivity measurements were obtained for each participant and a control group of about 375 employees in the same departments. The results indicated that subjects working under conditions 2 and 3 showed reductions in the extent and intensity of musculoskeletal health complaints, but not the subjects in condition 1. However, the subjects in condition 1 showed greater average improvement in productivity than those in conditions 2 and 3 and to the control group receiving no treatment. Of importance to designers was the finding that not all subjects showed improved productivity with the ergonomic improvements. In fact, about one-half of the subjects showed reduced productivity with the ergonomic improvements, even though the overall average for the ergonomic improvements showed a positive effect.

Gerr et al. (2002, 2005) and Marcus et al. (2002) prospectively examined the influences of computer use on workers' musculoskeletal and psychosocial health. They carried out ergonomics interventions to determine whether improvements in workstation design and employee postures could reduce musculoskeletal symptoms and disorders (Gerr et al. 2005). They conducted a randomized controlled trial of two different interventions among 376 computer users who used keyboards at least 15 hours per week and compared the subjects not receiving an intervention. They examined the incidence of neck/shoulder and hand/arm symptoms after implementation of the interventions for the intervention groups and control subjects. The two interventions consisted of adjusting each computer user's workstation for a better fit or training computer users to assume proper postures. The findings indicated that there were no significant differences among the two intervention groups and/or the comparison group. However, the findings also showed that the intervention groups complied with all components of the interventions for 24%–38% of the participants due to the inflexibility of workstation configurations. Thus, the majority of the employees in the intervention groups were able to achieve good ergonomic conditions.

Robertson (2007) investigated the effects of office ergonomic interventions on musculoskeletal health and group performance among knowledge workers at a corporate office in the United States. A new flexible office work environment was created for about 750 employees in an office building housing about 1750 employees. About 500 employees who were engaged in identical work but remained in traditional office workspaces in other parts of the building served as controls. The ergonomics interventions consisted of (1) a new flexible office space with adjustable workstations and flexible facility layout and (2) the new workstations plus office ergonomics training regarding the use of the new space and workstations. A total of 642 workers participated in the data collection, and 68% completed three rounds of data. Baseline musculoskeletal symptoms were taken with a follow-up at 3 and 6 months.

The flexible/adjustable workstation intervention group showed a 14% reduction in musculoskeletal symptoms from baseline to the 3-month measurements. But from the 3-month measurement to the 6-month measurement, there was a 19% increase in musculoskeletal symptoms. The flexible/adjustable workstation plus training intervention group showed a 48% reduction in musculoskeletal symptoms from baseline to the 3-month measurement, with a further reduction of 23% from the 3-month measurement to the 6-month measurement. There was no change in musculoskeletal symptoms for the control group over the 6-month measurement period. The flexible/adjustable workstation intervention group had an approximately 6% improvement in process time during the course of the experiment, whereas the workstation plus training intervention had an approximately 11% improvement and the control group had less than 1% improvement.

13.4 GUIDANCE FOR FIXED COMPUTER WORKSTATIONS

There are many online and hard-copy resources that provide specific guidance for computer users in setting up and using fixed computer workstations and work areas (ANSI/HFES-100 2007; BSR/HFES-100 2005; OSHA 2008; WorkSafeBC 2009; Smith, Carayon, and Cohen 2003, 2008). The guidance provided by these resources is generally consistent, and most provide specific information about each component of the workstation. We present below some general guidance for the design of fixed computer workstations and work areas without specific dimensions that are available in the aforementioned resources.

The recommended size of the work surface is dependent upon the task(s), documents, and technologies being used. The primary working surface (e.g., those supporting the keyboard, mouse, display(s), and documents) should be able to meet the following requirements:

- Allow the display screens to be moved forward or backward for a comfortable viewing distance for different employees with differing visual capabilities; provide means for adjusting the screen height to attain proper head/neck and back postures.
- Allow a detachable keyboard and a detachable mouse (or other pointing device) to be placed in several locations on the working surface to provide easy access and use. When using a laptop computer at a permanent fixed workstation, use a docking station to meet these requirements.
- Allow source documents to be positioned for easy viewing and proper musculoskeletal alignment of the upper extremities and the back when viewing the documents.
- Additional working surfaces (e.g., secondary working surfaces) may be necessary to store, lay out, read, write on, or manipulate documents, materials, or technologies (input devices, displays, computers, PIDs). These should provide easy access to materials and equipment on the surfaces, and comfortable postural shifts between the primary and secondary surfaces.
- Provide adequate knee and legroom for repositioning movements while working at primary and secondary surfaces.

- The tabletop should be as thin as possible for better thigh and knee clearance.
- Establish a comfortable table height that provides the necessary thigh/knee clearance, and allows input devices (keyboard, mouse) to be at comfortable heights. Adjustable tables provide the opportunity to better fit the users.
- Use keyboard trays for fixed height tables to enable better trunk and upper extremity postures when using the keyboard and mouse/trackball.
- Provide an adjustable height chair with swivel/tilt capabilities, adequate seat, and back padding and lumbar support.

Sometimes computer workstations are configured with multiple working surfaces so that several pieces of equipment and source materials can be easily accessible to the user. The setup should be arranged to allow for easy movement from one surface to another. Proper clearances under each working surface should be maintained, as well as a comfortable height to promote good posture of the back, neck/head, and upper extremities.

For all workstations, it is important to provide unobstructed room under the working surface(s) for the feet and legs so that users can easily shift their posture. Regular postural changes and body movement are important for reducing musculoskeletal fatigue and discomfort. Knee space height and width and toe depth are the three key factors for the design of clearance space under the working surfaces. Recommendations for clearance dimensions are provided in the ANSI/HFS-100, 2007 standard.

Table height has been shown to be an important contributor to musculoskeletal problems in computer users. Normal desk height of 30 in. (76 cm/30") is often too high for keyboard and mouse use by most people. It is desirable for the keyboard and mouse/trackball height to vary with the seated height of the user. A height adjustable keyboard/mouse/trackball tray provides this ability with a fixed height workstation. Height adjustable table surfaces can also achieve this, and adjustable multisurface tables encourage good posture by allowing the keyboard, mouse, trackball, and displays to be independently adjusted to appropriate keying, pointing, and viewing heights for each user and each task.

Tables that cannot be adjusted can be a problem when they are used by multiple users of differing sizes. When adjustable tables are used, the ease of making the adjustments is essential to encourage users to take the time to make the appropriate adjustments. Users should be instructed on how to adjust the workstation to be comfortable (Robertson 2007).

Adjustable tables allow vertical adjustments of the keyboard (and mouse) and displays. Some allow for the independent adjustment of the keyboard and display. Specifications for seated working surfaces' heights vary with whether the table is adjustable or at one fixed height and with a single working surface or multiple working surfaces (see ANSI/HFES 100-2007). For standing only workstations and for sit/stand workstations, the ANSI/HFES 100-2007 standard provides recommendations for the table surface height adjustment range.

13.5 DESIGN OF WORKSTATIONS FOR USE WITH LAPTOP COMPUTERS

Let us start with the laptop as a prime example of the influence of portability and efficiency, and then we will move on to other portable IT devices. A decade ago, the Human–Computer Interaction Committee of the International Ergonomics Association (IEA) produced a guideline for the use of laptop computers to improve ergonomic conditions (Saito et al. 2000). This was prompted by the ever-increasing sales of laptops and the replacement of fixed PCs by portable laptops on fixed working surfaces.

The primary advantage of the laptop is easy portability so the user can take the computer anywhere to do work. She or he can use it at the office, and then take it home to finish work, or take it with her or him to make a presentation at a meeting. The convenience and effectiveness of easy, lightweight portability are very high. In addition, all of the files are with the laptop, so nothing is mistakenly left behind at the office (or home). However, the comfort and health factors can be very low if the person uses the laptop in all manner of environments, workstations, and tasks that diminish the consistent application of good ergonomic principles. An important feature of the IEA laptop guideline (Saito et al. 2000) is to encourage situations of use that mirror the best practices of ergonomic conditions for fixed computer workstations. The best solution is to use a docking station for the laptop set up at a workstation that follows the guidance established for fixed workstations.

The following material in italic print has been extracted directly (with some minor editing) from the Saito et al. article "Ergonomic Guidelines for Using Notebook Personal Computers," which appeared in *Industrial Health*, volume 38, 2000, pp. 421–434.

> *Work Environment and Workstation Layout, "Create an environment that fits your work"*
>
> 1. *Use your laptop in a proper environment (lighting, temperature, noise, and so on). In particular, make sure the work area is neither too bright nor too dark.*
> 2. *Allocate enough space on your desk when placing a laptop.*
>
> *Chair and Desk, "Adjust chair height to match your physique"*
>
> 1. *Adjust your chair height based on the height of the keyboard, such that your forearm is parallel to the surface of the keyboard.*
> 2. *If your feet do not lie flat on the floor, provide a footrest.*
> 3. *Provide enough space underneath the desk.*
>
> *Keyboard, "Set the keyboard to a desirable angle, and use a palm rest if necessary"*

1. *Adjust the angle of the keyboard based on your posture and preferences.*
2. *Make sure there is space in front of the keyboard for you to comfortably rest your wrists (this space can be on the desktop surface itself if the keyboard is thin).*
3. *If the keyboard seems difficult to use, use an external keyboard.*

Working Posture, "Avoid unnatural postures, and change your posture occasionally"

1. *Avoid staying in postures where you are bent too far forward or backward, or twisted, for an extended duration.*
2. *Laptop users tend to view the display from too close, so make sure you maintain a distance of at least 40–50 cm between the display and your eyes.*
3. *Alternate near vision with far vision (i.e. observe object located at least 6 m far) as frequent as possible.*
4. *Make sure your wrists are not at an unnatural angle.*

Non-keyboard Input Devices, "Use a mouse as your pointing device if at all possible"

1. *If a mouse can be connected to your laptop, then do so as often as possible. Use a mouse pad whenever you use a mouse.*
2. *When you cannot connect a mouse, make sure you understand the built-in pointing device, and use the pointing device appropriately.*

These laptop guidelines provided by Saito et al. (2000) are useful when the laptop is used as a fixed PC at a docking station or at a desk (worktable). However, they do not provide as much help in the situations where there is no fixed workstation. We will describe some situations below where this can occur.

Imagine yourself sitting at the airport and your flight has been delayed for 2 hours. You have your laptop with you so you decide to get some work done while you wait. You could rent a cubicle or kiosk at the airport that would provide you with a high-speed Internet connection, a stationary telephone, a working surface (desk or table), a height adjustable chair, and some privacy (noise control, personal space). The characteristics of these work areas do not often conform to the best principles of ergonomic design. It is likely that the cubicle will provide some improvement over sitting with the laptop on your lap, but the characteristics may not meet the recommendations presented in this chapter. Such situations are acceptable for short exposures of up to 60 minutes, but longer exposures may lead to musculoskeletal discomfort, pain, and injury (if chronic). Now imagine that you have been told to stay in the boarding area because it is possible that the departure may be sooner than 2 hours. You get out your laptop, connect it to your cell phone, and place them on your lap. (That is why they are called laptops). You are sitting in a nonadjustable chair with poor back support.

This scenario is all too common. You can walk through O'Hare International Airport on any given day and see hundreds of people sitting at their boarding gates working on their laptops that are sitting on their laps. Now imagine a palm-held device that allows you to access your e-mail or to connect to the Internet. This device can be operated while you are standing in a line at the airport to check in, or sitting at the boarding gate like the laptop users. You can stand or sit punching at miniature buttons (sometimes with a stylus because they are so small) and interact with the interconnected world. Again, this scene is all too familiar in almost any venue (airport, restaurant, street, office).

In situations where there is not a fixed workstation, the device is typically positioned wherever is convenient. Very often such positioning creates bad postures for legs, back, shoulders, arms, wrists/hands, or neck. In addition, the smaller dimensions of the manual input devices (touch pad, buttons, keyboard, joy stick, roller ball) make motions much more difficult, and these often produce constrained postures and/or the use of too much force to operate the device. If the devices are used continuously for a prolonged period (such as 1 hour or more), muscle tension builds up, and discomfort in joints, muscles, ligaments, tendons, and nerves can occur. Some devices use voice/audio interfaces and can be used when you are walking (or even running). These might be headsets with earplugs and a microphone. These devices put additional load on the neck, and the voice/audio interfaces can strain the voice or the ears. These are "new" strains that we know very little about in terms of discomfort, health effects, and psychosocial effects. However, it is intuitive that frequent use of these interfaces will lead to strain and potential adverse consequences.

To reduce the undesirable effects of the poor workstation characteristics that lead to the discomfort, the following recommendations are given:

- If you are using a laptop on your lap, find a work area where you can put the laptop on a table (rather than on your lap). Then arrange the work area as closely as possible with the recommendations presented in the IEA laptop guidelines (Saito et al. 2000).
- If you are using a handheld PID, you should position yourself so that your back is supported. It is preferable to use the device sitting down. Of course, if you are using the PID as you are walking then this is not possible. If the PID has a voice interface, then use an earpiece and a microphone so that you do not have to hold it with your hand. But be sure not to overuse the device such that your voice and hearing are strained. Take frequent breaks from use to provide recovery for your voice and ears.
- Never work in poor postural conditions for more than 30 minutes continuously. Take at least a 5-minute break (preferably 10 minutes) away from the laptop/PID use, put the device down (away), get up and stretch for 1 minute or more, and then walk for 2–3 minutes (unless you are already walking in which case you should sit down for 5 minutes). If you are using a handheld PID in a standing position,

then during your break put it away, do 1 minute of stretching, and then sit down for 4 minutes. That may mean sitting on the floor, but preferably you will sit where you can support your back (against a wall, or a seat back).

- Buy equipment that provides the best possible input interfaces and displays (screens, headphones, typing pads). Since these devices are small, the perceptual-motor requirements for their use are much more difficult (sensory requirements, motion patterns, skill requirements, postural demands, and force demands). Therefore, screens should provide easily readable characters (large, understandable), and input buttons should be easy to operate (large, properly spaced, easily accessible, low force).
- Only use these devices when you do not have access to fixed workstations that have better ergonomic characteristics. Do not use these devices continuously for more than 30 minutes.

13.6 PORTABLE INFORMATION DEVICES, SMARTPHONES, AND CARRY-ALONG AND WEARABLE INFORMATION TECHNOLOGY

We have already discussed some ergonomic issues of using PIDs. We can foresee an even greater potential for ergonomic concerns with the use of PIDs than with laptop computers. PIDs are made to be as small and light as possible for portability and convenience of carrying. This is good as the lower weight produces smaller loads of the body. But small-sized devices are more difficult to manipulate with the hands and to observe displays with the eyes. This has led many designers to emphasize verbal/auditory and haptic interfaces that do not require substantial manipulation by the hands or good vision (Hirose and Hirota 2005; HCII 2005). In fact, several new PIDs have the capability to communicate with other IT devices without human intervention (HCII 2005, 2007, 2009, 2011).

Now let us explore some current applications of PIDs and some possible future applications to see where ergonomic and workstation issues might emerge. There are millions and millions of cell phones in use worldwide, and cell phones are a good representative of the PID. There is virtually nowhere in the world where cell phone cannot be used (with a few exceptions), and there is virtually no one (with adequate finances) who cannot access a cell phone and connect with the world. Cell phones can have many capabilities including telephoning, e-mailing, texting, auditory streaming, Internet surfing, walkie-talkie communicating, photographing, video camera picturing and recording, and television/cable/satellite broadcast receiving. Even with all of these built-in features, the size of cell phones is shrinking. As cell phones shrink, the manual hand interfaces are getting smaller, as are the visual displays. With their small size, cell phones can be carried easily and can be used when a person is walking, sitting, running, lying down, or hanging upside down. So what are the ergonomic concerns with their use, and in particular, the workstation design issues?

Small manual interfaces and displays make the accurate and comfortable application of perceptual-motor skills difficult (Albers and Kim 2002; Haggerty and Tarasewich 2005; Myers and Wobbrock 2005; Hinckley 2008). A cell phone can be held in one hand (this hand becomes the workstation) and then be manipulated with the other hand. In some instances, the hand holding the cell phone is also used to manipulate the manual interface. Either of these situations leads to workstation conditions where the users cannot apply their highest level of perceptual-motor skills. For example, they cannot use both hands for inputting into the interface, or the device is held at an awkward angle for manual inputting, or the posture of the trunk is unstable, which limits the capacity to effectively use the hands.

Due to this, a preferred form of input to control the action of the PID is speech (Lai, Karat, and Yankelovich 2008), whereas the displays are typically a combination of visual and auditory information. These interfaces may lead to ergonomic problems of overuse of the voice, increased duration of mental concentration, increased eye strain and visual discomfort, and increased error rates of the communication with the devices. There is no body of research data to tell us if these problems are or will become prevalent among PID users, or whether there will be long-term effects on comfort, health, and performance. For now, we can only conjecture about the possibilities for problems. But it is clear that many people are using PIDs and other computer-based technologies many hours per day, and this increased extent of use will likely lead to discomfort and some health effects for voice, ears, upper extremities, and mental stress.

Now let us think about the workstation issues with PIDs using cell phone as an example. We will go back to the airport example, and I will use my cell phone to communicate with my office. At this time, I am standing in a long line at the airline check-in counter. My cell phone has voice activation and control capability. I ask my cell phone for my e-mail service provider and up come my e-mails on the display screen. I am using my thumb to scroll through the e-mails one at a time. I hold the cell phone close to my face to enhance my ability to read the display screen. I shuffle forward as the line moves toward the airline counter (automated check-in station). I am standing, and I have no postural support for my back, buttocks, and legs. It is true that I would be standing in the line feeling the postural strain even if I were not using my cell phone. But my cell phone use adds extra postural loading since I am manually (or vocally) manipulating the interface to operate the e-mail processing. The small visual display may create eyestrain, and if I talk on the cell phone for too long it may create voice strain.

One way to reduce the load would be if I had a workstation where I could sit and support my back, buttocks, and legs, even be able to rest my arms and elbows as I manually manipulate the interface. I could use a vocal/auditory interface with a head-mounted earplug and microphone (headset) and thus eliminate the manual input and visual display viewing. The headset is a workstation improvement that reduces the manual manipulation and some of the postural loading on the upper extremities and back. But the headset adds some weight to the head, and this increases loading on the neck, shoulders, and back. One difficulty with using an auditory interface is the high ambient noise in the airport lounge that interferes with and masks the cell phone's auditory signals. To reduce this auditory interference, a helmet with acoustical privacy could serve as a workstation improvement. However, a helmet adds substantial weight to the head, which puts increased loading on the neck, shoulders, and back.

A major workstation improvement for users waiting in a line, which would reduce the postural loading when using a cell phone, is to provide chairs so a user can sit down. A chair provides postural support for the back, buttocks, and legs. A chair on casters/wheels that can be scooted along as the line moves forward could be beneficial. But this could increase the congestion in the lounge area as people are moving through the line. This one example illustrates the new loads that may be added to technology users' lives by the use of PIDs. We reiterate some of the advice from Section 13.5 as this is also applicable to the use of PIDs and other portable devices.

To reduce the undesirable effects of a lack of a workstation that can lead to the users' discomfort, the following recommendations are given:

If you are using a handheld PID, you should position yourself so that your back is supported. It is preferable to use the device sitting down. Of course, if you are using the PID as you are walking, then this is not possible. If the PID has a voice interface, then use an earpiece and a microphone so that you do not have to hold it with your hand. But be sure not to overuse the device such that your voice and hearing are strained. Take frequent breaks from use to provide recovery for your voice and ears.

Never work in poor postural conditions for more than 30 minutes continuously. Take at least a 5-minute break (preferably 10 minutes) away from the laptop/PID use, put the device down (away), get up and stretch for 1 minute or more, and then walk for 2–3 minutes (unless you are already walking in which case you should sit down for 5 minutes). If you are using a handheld PID in a standing position, then during your break put it away, do 1 minute of stretching, and then sit down for 4 minutes. That may mean sitting on the floor, but preferably you will sit where you can support your back (against a wall or a seat back).

Buy equipment that provides the best possible input interfaces and displays (screens, headphones, typing pads). Because these devices are small, the perceptual-motor requirements for their use are much more difficult (sensory requirements, motion patterns, skill requirements, postural demands, and force demands). Therefore, screens should provide easily readable characters (large, understandable), and input buttons should be easy to operate (large, properly spaced, easily accessible, low force).

Do not use these devices continuously for more than 30 minutes.

13.7 CHAIR AS A CRITICAL ELEMENT OF THE WORKSTATION

It was not until the last 40 years that sitting posture and chairs (seats) became topics for scientific research, especially for ergonomics and orthopedics. Studies have revealed that the sitting position, compared with the standing position, reduces static muscular efforts in legs and hips, but increases the physical load on the intervertebral discs in the lumbar region of the spine.

The debate over what constitutes proper seated posture is not yet fully resolved. Is an upright-seated posture most healthy, or is a relaxed posture with a backward-leaning trunk healthier? Interesting experiments by the Swedish surgeons Nachemson and Elfstrom (1970) and Andersson and Ortengreen (1974) offer some guidance about this. These authors measured the pressure inside the intervertebral discs and the electrical activity of the back muscles in relation to different sitting postures. When the backrest angle of the seat was increased from 90° to 120°, subjects showed a significant decrease in the intervertebral disc pressure and the electromyographic activity of the back. Since heightened pressure inside the intervertebral discs means that they have more stress, the authors concluded that a sitting posture with reduced disc pressure is more healthy and desirable.

Most ergonomic standards for computer workstations are based on a more traditional view about a healthy sitting posture. Mandal (1982) reported that the "correct seated position" goes back to 1884 when the German surgeon Staffel recommended the well-known upright position. Mandal (1982) stated:

> *But no normal person has ever been able to sit in this peculiar position (upright trunk, inward curve of the spine in the lumbar region, and thighs in a right angle to the trunk) for more than 1–2 minutes, and one can hardly do any work as the axis of vision is horizontal. Staffel never gave any real explanation why this particular posture should be better than any other posture. Nevertheless, this posture has been accepted ever since quite uncritically by all experts all over the world as the only correct one.*

It is our observations over many years of field research and consultation that the sitting posture of computer users is very seldom an upright position of the trunk for extended periods. When seated at a computer, some people lean backward, whereas others lean forward and some sit up straight. Most people shift sitting positions when there is fatigue, discomfort, or pain in the musculoskeletal system. Based on our experience, we believe that an important consideration for seated posture is to have workstation and chair designs that allow for a variety of seated postures and movement of the computer users while they are seated.

Chair adjustability in terms of vertical height of the seat pan, the seat pan angle, and providing lumbar support help to promote trunk, shoulder, neck, and leg postures that reduce strain on the muscles, tendons, ligaments, and discs. The postural support and movement action of the chair help maintain proper seated posture and encourage good movement patterns. A chair that provides swivel action encourages movement, whereas backward tilting increases the number of postures that can be assumed.

The chair height should be adjustable so that the computer operator's feet can rest firmly on the floor with minimal pressure beneath the thighs. The minimum range of adjustment for seat pan height is provided in the ANSI/HFES-100 standard (2007).

To enable short users to sit with their feet on the floor without compressing their thighs, it may be necessary to add a footrest. A well-designed footrest has the following features:

- It is inclined upwards slightly (about 5°–15°).
- It has a nonskid surface.
- It is heavy enough that it does not slide easily across the floor.
- It is large enough for the feet to be firmly planted.
- It accommodates persons of different stature.

The seat "pan" is where the person sits on the chair. It is the part of the chair that directly supports the weight of the buttocks. The seat pan should be wide enough to permit operators to make slight shifts in posture from side to side. This not only helps to avoid static postures, but also accommodates a large range of individual buttock sizes. The seat pan should not be overly U-shaped because this can lead to static sitting postures. The seat pan dimensions are provided in the ANSI/HFES-100 standard (2007). The front edge of the seat pan should be well rounded downward to reduce pressure on the underside of the thighs that can affect blood flow to the legs and feet. This feature is often referred to as a "waterfall" design. The seat needs to be padded to the proper firmness that ensures an even distribution of pressure on the thighs and buttocks. A properly padded seat should compress about 1/2 to one inch when a person sits on it.

Some experts feel that the seat front should be elevated slightly (up to 7°), whereas others feel it should be lowered slightly (about 5°). There is some disagreement among the experts about the correct answer, and due to this disagreement, many chairs allow for both front and backward angling of the front edge of the seat pan. The operator can then angle the chair's front edge to a comfortable position. The ANSI/HFES-100 standard provides information about the adjustability range of the seat pan front edge. The seat pan height and angle adjustments should be accessible and easy to use from a seated position.

The tension and tilt angle of the chair's backrest should be adjustable. Inclination of chair backrest is important for operators to be able to lean forward or back in a comfortable manner while maintaining a correct relationship between the seat pan angle and the backrest inclination. A backrest inclination of about 110° is considered an appropriate posture by many experts. However, studies have shown that operators may incline backwards as much as 125°, which also is an appropriate posture. Backrests that tilt to allow an inclination of up to 125° are therefore a good idea. The backrest tilt adjustments should be accessible and easy to use. An advantage of having an independent tilt angle adjustment is that the backrest tilt will then have little or no effect on the front seat height or angle. This also allows operators to shift postures readily. The ANSI/HFES-100 standard (2007) provides advice about back rest adjustment ranges.

Chairs with high backrests are preferred since they provide support to both lower back and the upper back (shoulder). This allows employees to lean backward or forward, adopting a relaxed posture and resting the back and shoulder muscles. The ANSI/HFES-100 standard (2007) provides guidance on the design of the seat backrest. To prevent back strain, it is also recommended that chairs have lumbar (mid-back) support, since the lumbar region is one of the most highly strained parts of the spine when sitting.

For most computer workstations, chairs with rolling castors or wheels are desirable: These promote movement and facilitate postural adjustment, particularly when the operator has to reach for equipment or materials that are on the secondary working surfaces. Chairs should have five supporting legs.

Another important chair feature is armrests. Both pros and cons to the use of armrests at computer workstations have been advanced. On the one hand, some chair armrests can present problems of restricted arm movement, interference with the operation of input devices, pinching of fingers between the armrest and table, restriction of chair movement, such as under the work table, irritation of the arm or elbows due to tissue compression when resting on the armrest, and adoption of awkward postures. Properly designed armrests can overcome the problems mentioned above. Armrests can provide support for resting the arms to prevent or reduce arm, shoulder, and neck fatigue. Removable armrests are an advantage because they provide greater flexibility for individual operator preference. For specific tasks such as using a numeric keypad, a full armrest can be beneficial in supporting the arms. Many chairs have height adjustable armrests that are helpful for operator comfort, and some allow for adjusting the angle of the armrests as well.

13.8 ADDITIONAL WORKSTATION CONSIDERATIONS

Providing the capability for the screen to swivel and tilt up/down gives the user the ability to better position the screen for easier viewing. Reorientation of the screen around its vertical and horizontal axes can help to position a screen to reduce screen reflections and glare. Reflections can be reduced by simply tilting the display slightly back or down, or to the left or right away from the source of glare. The perception of screen reflections depends not only upon screen tilt, but also upon the operator's line of sight.

An important component of the workstation that can help reduce musculoskeletal loading is a document holder. When properly designed, proportioned, and placed, document holders reduce awkward inclinations of the head and neck and frequent movements of the head up and down and back and forth. They permit source documents to be placed in a central location at the same viewing distance as the computer screen. This eliminates needless head and neck movements and reduces eyestrain. In practice, some flexibility about the location, adjustment, and position of the document holder should be maintained to accommodate both task requirements and operator preferences. Dainoff (1982) showed the effectiveness of an in-line document holder. The document holder should have a matte finish so that it does not reflect light.

Privacy requirements include both visual and acoustical control of the workplace. Visual control prevents physical intrusions, contributes to confidential/private conversations, and prevents the individual from feeling constantly watched. Acoustical control prevents distracting and unwanted noise (from machine or conversation) and permits speech privacy. Although certain acoustical methods and materials such as freestanding panels are used to control general office noise level, they can also be used for privacy. Planning for privacy should not be made at the expense of visual interest or spatial clarity. For instance, providing wide visual views can prevent the individual from feeling isolated. Thus, a balance between privacy and openness enhances user comfort, work effectiveness, and office communications. Involving employees in decisions of privacy can help in deciding the compromises between privacy and openness.

The use of a wrist rest when keying can help to minimize extension (backward bending) of the hand/wrist, but the use of a wrist rest for operators' comfort and health has generated some debate because there are trade-offs between comfort and health. When the hand or wrist is resting on the wrist rest, there is compression of the tissue that may create increased carpal canal pressure or local tissue ischemia. On the other hand, the wrist rest allows the hands and shoulders to be supported with less muscular tension, which is beneficial to computer operator comfort. At this time, there is no scientific evidence that the use of a wrist rest either causes or prevents serious musculoskeletal disorders of the hands, wrists, or shoulders. Thus, the choice to use a wrist rest should be based on employees' comfort and performance considerations until scientific evidence suggests otherwise.

If used, the wrist rest should have a fairly broad surface (5 cm minimum) with a rounded front edge to prevent cutting pressure on the wrist and hand. Padding further minimizes skin compression and irritation. Height adjustability is important so that the wrist rest can be set to a preferred level in concert with the keyboard height and slope.

Arm holders are also available to provide support for the hands, wrists, and arms while keyboarding and have shown to be useful for shoulder comfort. The placement of the arm holder should not induce awkward postures in its use. The device should be placed within easy reach of the operator, especially when it will be used frequently during work.

When keyboard trays are used, they should allow for the placement of other input devices directly on the tray instead of on other working surfaces.

13.9 VISUAL ENVIRONMENT

Computers and PIDs have screens that are susceptible to poor viewing due to glare from high-illumination sources and particles on the screen. For instance, luminance sources in the environment can fall on the screen and wash out characters on the screen, or the accumulation of dust and/or dirt particles on the screen can block or distort images. These conditions not only affect the ability to read the screen, but can also lead to visual fatigue and dysfunction. Specific characteristics of the environment such as illumination level and glare have been related to visual strain problems of a computer operator.

The alignment of lighting in relation to the computer workstation, as well as levels of illumination in the area surrounding a computer workstation, has been shown to influence the ability of the computer operator to read hard copy and the computer. Readability is also affected by the differences in luminance contrast in the work area. The level of illumination affects the extent of reflections from working surfaces and from the screen surface. Mismatches in these characteristics as well as the nature of the job tasks are believed to cause the visual system to be overworked, which can lead to visual fatigue and discomfort. Boyce (2006) indicates that improper illumination can affect performance, visual discomfort, fatigue, and mood.

Proper illumination and glare reduction are important aspects of the visual environment that influence computer and PID screen and hard-copy readability and viewing in the general environment. The illumination required for a particular task is determined by the visual requirements of the task and the visual ability of the employees doing the task. High levels of illumination are generally bad for reading from computer and PID screens, but good for reading hard copy. Lower levels of illumination will provide better computer and PID screen image quality and reduced screen glare. Illuminance in the range of 300–700 lux measured on the horizontal working surface (not the computer screen) is normally preferable. The lighting level should be set up according to the visual demands of the tasks performed. For instance, higher illumination levels are necessary to read hard copy and lower illumination levels are better for work that just uses the computer and PID screen. Thus, a job in which hard copy and a computer screen are both being used should have a general work area illumination level of about 500–700 lux; and a job that only requires reading the computer screen would have a general work area illumination of 300–500 lux.

Conflicts can arise when both hard copy and computer screens are used by different employees who have differing job task requirements or differing visual capabilities and are working in the same room. As a compromise, room lighting can be set at the lower level (300 lux) or intermediate level (500 lux), and additional task lighting for the hard copy tasks can be provided at each workstation as needed. Such additional lighting must be carefully shielded and properly placed to avoid glare and reflections on the computer screens and adjacent working surfaces of other employees. Furthermore, task lighting should not be overly bright in comparison with the general work area lighting, since the contrast between these two different light levels may produce eyestrain.

The surface of the computer and PID screens reflects light and images from the environment. The luminance of the reflections decreases character contrast and disturbs legibility; it can be so strong that it produces a glare. Image reflections are annoying, especially since they also interfere with focusing mechanisms; the eye is induced into focusing between the text and the reflected image. Thus, reflections are also a source of distraction.

Luminance is a measure of the brightness of a surface, the amount of light leaving the surface of an object, either reflected by the surface (as from a wall or ceiling), emitted by the surface (as from the screen characters), or transmitted (as light from the sun that passes through translucent curtains). High-intensity luminance sources (such as windows) in the peripheral field of view should be avoided. In addition, a balance among luminance levels within the computer user's field of view should be maintained. To reduce environmental glare, the luminance ratio within the user's near field of vision should be approximately 1:3, and approximately 1:10 within the far field of vision. For luminance on the screen itself, the character-to-screen background luminance contrast ratio should be at least 7:1. To give the best readability for each operator, it is important to provide screens with adjustments for character contrast and brightness. These adjustments should have controls that are obvious and easily accessible from the normal working position (e.g., located at the front of the screen).

Experts have traditionally recommended a viewing distance between the screen and the operator's eye of 45–50 cm, but no more than 70 cm. However, experience in field studies has shown that users may adopt a viewing distance greater than 70 cm or lesser than 45 cm and still be able to work efficiently and comfortably. Thus, viewing distance should be determined in context with other considerations. It will vary depending upon the task requirements, computer or PID screen characteristics, and an individual's visual capabilities. For instance, with poor screen quality and poor vision, it may be necessary to reduce viewing distance for easier character recognition. Typically, the viewing distance will be 50 cm or less because of the small size of the characters on the computer or PID screen.

13.10 AUDITORY ENVIRONMENT

As the use of PIDs in a wide variety of environments increases the need for a quiet auditory environment that allows the user to easily hear and speak at a normal loudness level becomes more important. Crowded, noisy environments detract from users' ability to hear. Many PIDs have auditory interfaces such as headphones, which can concentrate the primary auditory signal and block out some of the environmental noise. One concern is that users may increase the loudness beyond levels that are safe for the auditory sensory system due to the ambient noise in the environment or for personal preference (CDC 2003). High ambient noise in the environment leads to the need for greater mental concentration (attention) to the auditory signals, increased intensity of the primary auditory signal (up to levels that may cause temporary auditory threshold shifts), and mood disturbances (irritation, anger, discouragement) (Casali 2006). These effects can result in increased mental fatigue, psychological stress; and long-term exposures may lead to discomfort and health consequences.

Although headphones may provide benefits to concentrate the primary auditory signal, they also have the drawback of providing too much sound energy to the ears. Prolonged exposure to loud primary auditory signals can cause the adverse effects described above. To protect the auditory sensory system when using PIDs, we proposed the following:

Find environments to use your PID where the ambient auditory levels are 40–50 dba or less.

Do not use your PID for more than 30 minutes without taking a break where you can rest your eyes and ears. You should rest for at least 10 minutes before you start using your PID again. As your total use of the PID increases over the course of a day, then the rest breaks should become longer. Although the literature suggests that taking a break after 30 minutes of PID use should be sufficient for your eyes to recover (given a proper visual environment as defined above), there is no literature that provides guidance on the maximum amount of time of PID use before a break is necessary to provide recovery for your ears and mental mood. Nor is there literature to indicate the minimum amount of break time necessary to achieve auditory or mental recovery. Thus, we suggest a conservative approach of 15 minutes of rest away from loud sound after 30 minutes of exposure.

Keep the volume of the headphones of the PID at the lowest level necessary to hear the primary auditory signal properly. In no instance exceed 85 dba, and do not exceed 30 minutes of continuous exposure (see the above point). It is best to not exceed 70 dba, and if you adhere to 1 above, then you will not have to exceed 60–70 dba if you have normal hearing.

Be courteous when using your microphone, cell phone, and PID. Do not talk directly at other people such that your conversation may produce masking noise that interferes with their conversations or peace. Keep your voice as low as possible to allow your interface or listener to understand your signal.

13.11 ERGONOMIC IMPLEMENTATION ISSUES

Implementing a workplace change, such as improving workstation design or work methods, is a complex process because it impacts many elements of the work system (Derjani-Bayeh and Smith 1999; Smith and Carayon 1995; Hagberg et al. 1995; Smith and Sainfort 1989; Carayon and Smith 2000; Smith and Derjani-Bayeh 2003; Robertson 2007). Managers, designers, and engineers often like to believe that technological enhancements are easy to make and that performance and health improvements will be immediate and substantial. Proper implementation involves changes in more than the workstation, for instance, the work organization, job content, task improvements, job demands, training, and socialization issues need to be considered. Planning for change can help the success of implementation and reduce the stress generated by the change. But the success of implementing change depends heavily on the involvement and commitment of the concerned parties, in particular management, technical and support staff, first-line supervision, and employees.

There is universal consensus among change management experts that the most successful strategies for workplace improvements involve all elements (subsystems) of the work system that will be affected by the change (Hendrick 1986; Lawler 1986; Smith and Carayon 1995; Carayon and Smith 2000). Involvement assumes that there is an active role in the change process, not just providing strategic information. Active participation generates greater motivation and better acceptance of solutions than passively providing information and taking orders. Active participation is achieved by soliciting opinions and sharing authority to make decisions about solutions. However, one drawback of active participation is the need to develop consensus among participants who have differing opinions and motives. This usually takes more time than traditional decision making and can bring about conflict among subsystems. Another drawback is that line employees often do not always have the technical expertise necessary to form effective solutions.

Participative ergonomics can take various forms, such as design decision groups, quality circles, and worker-management committees. Some of the common characteristics of these various programs are employee involvement in developing and implementing ergonomic solutions, dissemination and exchange of information, pushing ergonomics expertise down to lower levels, and cooperation between experts and nonexperts. One of the characteristics of participatory ergonomics is the dissemination of information (Hendrick 1986; Noro 1991; Carayon and Smith 2000). Participative ergonomics can be beneficial to reduce or prevent resistance to change because of the information provided to the various members of the organization concerned with the new technology. Uncertainty and lack of information are two major causes of resistance to change and have been linked to increased employee stress. If employees are informed about potential ergonomics changes in advance, they are less likely to actively resist the change.

Training computer users about how the new workstation functions and operations are important especially if the adjustment controls are neither obvious nor intuitive. Hagberg et al. (1995) have indicated that employee training is a necessary component to any ergonomic program for reducing work-related musculoskeletal disorders. Green and Briggs (1989) and Robertson (2007) found that adjustable workstations are not always effective without appropriate information about benefits of adjustments and training in how to use the equipment. Hagberg et al. (1995) suggested the following considerations for ergonomics training programs:

Have employees involved in the development and process of training. Using employees' work experiences can be helpful in illustrating principles to be learned during training. In addition, using employees as instructors can be motivational for the instructors and learners.

Use active learning processes where learners participate in the process and apply "hands-on" methods of knowledge and skill acquisition. This approach to learning enhances acquisition of inputs and motivation to participate.

Apply technology to illustrate principles such as audiovisual equipment and computers. Much like active processes, technology provides opportunities for learners to "visualize" the course materials and to test their knowledge dynamically and immediately.

Use of on-the-job training is preferred over classroom training. Both can be effective when used in combination.

13.12 APPLICATIONS EXAMPLE: COMPUTER WORKSTATIONS IN HEALTH-CARE SETTINGS

There is a major push toward implementation of various forms of computer technology in health-care settings. Through the Recovery Act, about $19 billion is being invested in computerized medical records (http://www.whitehouse.gov/issues/health-care). The objective is to use health information technology (or health IT such as electronic medical records) to improve health-care quality and patient safety, and reduce health-care costs. Much attention has been paid to the design of health IT interface such as usability and fit of the technology with the cognitive work of health-care providers (Stead and Lin 2009); however, the issues of workstation design and physical ergonomics have been largely ignored. Given the extensive musculoskeletal problems experienced by nurses and other health-care professionals (Feyer et al. 2000; Hignett 2007), the additional physical and psychosocial workload associated with computer use may exacerbate health and safety problems.

Carayon et al. (2009) conducted an in-depth case study of the implementation of an electronic health records (EHR) technology in a small clinic. Various types of data were collected before and after the EHR implementation to assess the impact of the technology on work, working conditions, and outcomes. A work analysis showed that the amount of time spent on computer work significantly increased after the EHR implementation: from about 2% to 21% for physicians, from 4% to 11% for other clinical staff, and from 19% to 31% for administrative staff. Survey data collected from physicians, nurses, other health-care professionals, and administrative staff at the clinic showed small increase in perceived workload and a decrease in amount of control over resources. Clinic staff also reported to be significantly more dependent on computers to get their work done after the EHR implementation. With regard to health outcomes, there was a slight increase in the percentage of clinic staff who reported back pain and pain or stiffness in arms or legs, and a slight decrease in terms of swollen or painful muscles and joints. This single case study of EHR implementation in a small clinic shows that the use of computer work can produce some negative impact, such as increased workload and increased self-reported musculoskeletal problems. However, we do not know the long-term effects of EHR implementation; the data collected in the study by Carayon et al. (2009) provided information about the effect of the technology only after 1 year. In addition, we know very little about workstation design for health IT in other health-care settings, such as hospitals.

Health IT can be implemented in a range of hardware configurations, such as stationary computers, computers on wheels (COW), or other mobile configurations (e.g., notebook or Tablet PC). Andersen et al. (2009) conducted a study of computer hardware device in two hospital units in Australia. Four different hardware configurations were available: (1) stationary computer stations, (2) laptops mounted on trolleys, (3) an ergonomically designed computer on wheel (i.e., integrated computer and cart device specifically designed to be easy to use), and (4) tablet PCs. Observations and interviews were performed to understand the way nurses and physicians used the various technologies; in addition, two researchers assessed the physical characteristics and usability of the technologies. The COWs and tablet PCs were used most frequently in the patients rooms (about 57% of time) and in the corridors (about 36%). Nurses tended to prefer to use the generic COW instead of the ergonomic COW. Nurses infrequently used stationary computers to perform their tasks. On the other hand, physicians were more likely to use stationary computers, most often within their offices. When doing rounds in the hospital, physicians used primarily generic COWs (about 57% of time) and tablet PCs (about 36% of time). All nurses reported to prefer to use a generic COW over other devices; physicians used both tablet PCs and generic COWs during rounds. Nurses and physicians reported that generic COWs were easier to use because the trolley had space for storing medications, paper documents, and other equipment. Even though the ergonomic COW was better designed from a physical ergonomics viewpoint, it did not support the noncomputer tasks performed by nurses and physicians (e.g., use of paper documents, storing medications). The mobility of computer devices was a very important factor for both nurses and physicians. They perform work under high time pressure; therefore, having easy access to mobile computers is critical.

An important aspect of computer workstation design in health-care settings is the need for collaborative activities. The study by Andersen et al. (2009) shows that many computer tasks are performed in a collaborative manner, such as nurses collaborating with other nurses and student nurses and physicians on rounds collaborating with other physicians and medical students. Therefore, the design of the computer work areas needs to support the collaborative activities performed by health-care professionals. There needs to be sufficient room for several people to use computers simultaneously or work together on a single computer. Most hospital units have not been designed to accommodate the new technologies (e.g., COWs) and the new forms of work organization (i.e., teamwork and collaboration); this may lead to a range of physical and psychosocial problems that can affect the performance of health-care professionals, therefore affecting the quality and safety of care provided to patients.

13.13 SUMMARY OF RECOMMENDATIONS

A fixed computer workstation (work area) comprises a computer, input and output interfaces, the furniture where the computer is used, and the physical environment in which the computer is used. The design of these elements and how they fit together play a crucial role in users' performance and in minimizing potential adverse discomfort and health consequences. The recommendations presented in this chapter address the physical environment and implementation issues. Organizational factors and task-related factors should also be taken into account as they affect and/or depend on technology users. PIDs require creative work area and workstation solutions to achieve the same effectiveness of fixed computer workstations. PIDs pose unique problems, and users need to take actions that will minimize loads on their musculoskeletal system, sensory systems, and mental processes. PIDs have inherent ergonomic problems due to their ubiquitous applications, small size, and potential for users' overexposure to prolonged use. Using PIDs in the proper environment, limiting the extent of use by taking frequent rest breaks, and having the best possible interfaces is good ergonomic sense.

REFERENCES

Albers, M., and L. Kim. 2002. Information design for small-screen interface: An overview of web design issues for personal digital assistants. *Tech Commun* 49(1):45–60.

Andersen, P., A.-M. Lindgaard, M. Prgomet, N. Creswick, and J. I. Westbrook. 2009. Mobile and fixed computer use by doctors and nurses on hospital wards: Multi-method study on the relationships between clinician role, clinical task, and device choice. *J Med Int Res* 11(3):e32.

Andersson, B. J. G., and R. Ortengreen. 1974. Lumbar disc pressure and myoelectric back muscle activity. *Scand J Rehabil Med* 3:115–21.

ANSI/HFES. 1988. *American National Standard for Human Factors Engineering of Visual Display Terminal Workstations* (ANSI/HFS Standard No. 100-1988). Santa Monica, CA: The Human Factors Society.

ANSI/HFES. 2007. *American National Standard for Human Factors Engineering of Computer Workstations* (ANSI/HFS Standard No. 100-2007). Santa Monica, CA: The Human Factors Society.

Bergqvist, U. O. 1984. Video display terminals and health: A technical and medical appraisal of the state of the art. *Scand J WorkEnviron Health* 10(Suppl 2):87.

Bernard, P. B. 1997. *Musculoskeletal Disorders and Workplace Factors DHHS (NIOSH) Publication No. 97-141*. Washington, DC: National Technical Information Service.

Boyce, P. 2006. Illumination. In *Handbook of Human Factors and Ergonomics,* 3rd ed., ed. G. Salvendy, 643–69. Hoboken, NJ: John Wiley & Sons, Inc.

BSR/HFES. 2005. Human *Factors Engineering of Computer Workstations (BSR/HFES-100)*. Santa Monica, CA: The Human Factors and Ergonomics Society.

Cakir, A., D. J. Hart, and T. F. M. Stewart. 1979. *The VDT Manual.* Darmstadt, Germany: Inca-Fiej Research Association.

Carayon, P., and M. J. Smith. 2000. Work organization and ergonomics. *Appl Ergon* 31:649–62.

Carayon, P., M. J. Smith, and M. C. Haims. 1999. Work organization, job stress, and work-related musculoskeletal disorders. *Hum Factors* 41:644–63.

Carayon, P., P. Smith, A. S. Hundt, V. Kuruchittham, and Q. Li. 2009. Implementation of an electronic health records system in a small clinic. *Behav Inf Technol* 28(1):5–20.

Casali, J. G. 2006. Sound and noise. In *Handbook of Human Factors and Ergonomics*, 3rd ed., ed. G. Salvendy, 612–42. Hoboken, NJ: John Wiley & Sons, Inc.

CDC. 2003. *Third National Health and Nutrition Examination Survey (NHANES lll) 1988–94*. Atlanta, GA: Centers for Disease Control. www.cdc.gov.

Cohen, W. J., C. A. James, A. D. Taveira, B. Karsh, J. Scholz, and M. J. Smith. 1995. Analysis and design recommendations for workstations: A case study in an insurance company. In *Proceedings of the Human Factors and Ergonomics Society 39th Annual Meeting*, 1, 412–6. San Diego, CA: Human Factors and Ergonomics Society.

Dainoff, M. J. 1982. Occupational stress factors in visual display terminal (VDT) operation: A review of empirical research. *Behaviour and Information Technology* 1(2):141–76.

Derjani-Bayeh, A. and M. J. Smith. 1999. Effects of physical ergonomics on VDT workers' health: A longitudinal intervention study in a service organization. *International Journal of Human Computer Interaction* 11(2):109–35.

Eltayeb, S. M., J. B. Staal, A. A. Hassan, S. S. Awad, and R. A. de Bie. 2008. Complaints of the arm, neck and shoulder among computer office workers in Sudan: A prevalence study with validation of an Arabic risk factor questionnaire. *Environ Health* 7:33–.

Eltayeb, S., J. B. Staal, A. Hassan, and R. A. de Bie. 2009. Work related risk factors for neck, shoulder and arm complaints: A cohort study among Dutch computer office workers. *J Occup Rehabil* 19(4):315–22.

Eltayeb, S., J. B. Staal, J. Kennes, P. H. Lamberts, and R. A. de Bie. 2007. Prevalence of complaints of arm, neck and shoulder among computer office workers and psychometric evaluation of a risk factor questionnaire. *BMC Musculoskelet Disod* 8:68–.

Feyer, A. M., P. Herbison, A. M. Williamson, I. de Silva, J. Mandryk, L. Hendrie et al. 2000. The role of physical and psychological factors in occupational low back pain: A prospective cohort study. *Occup Environ Med* 57(2):116–20.

Gerr, F., M. Marcus, C. Ensor, D. Kleinbaum, S. Cohen, A. Edwards, E. Gentry, D. J. Ortiz, and C. Monteilh. 2002. A prospective study of computer users: I. Study design and incidence of musculoskeletal symptoms and disorders. *Am J Ind Med* 41(4):221–35.

Gerr, F., M. Marcus, C. Monteilh, L. Hannan, D. Ortiz, and D. Kleinbaum. 2005. A randomized controlled trial of postyral interventions for prevention of musculoskeletal symptoms among computer users. *Occup Environ Med* 62(7):478–87.

Grandjean, E. 1979. *Ergonomical and Medical Aspects of Cathode Ray Tube Displays*. Zurich, Switzerland: Federal Institute of Technology.

Grandjean, E. 1980. Ergonomics of VDUs: Review of present knowledge. In *Ergonomics Aspects of Visual Display Terminals*, ed. E. Grandjean and E. Vigliani, 1–12. London, England: Taylor & Francis, p. 1–12.

Grandjean, E. 1987. Design of VDT workstations. In *Handbook of Human Factors*, ed. G. Salvendy, 1359–97. New York: John Wiley and Sons.

Green, R. A., and C. A. Briggs. 1989. Effect of overuse injury and the importance of training on the use of adjustable workstations by keyboard operators. *J Occup Med* 31:557–62.

Gunnarsson, E., and O. Ostberg. 1977. *Physical and Emotional Job Environment in a Terminal-Based Data System* (1977:35). Stockholm, Sweden: Department of Occupational Safety, Occupational Medical Division, Section for Physical Occupational Hygiene.

Hagberg, M., B. Silverstein, R. Wells, M. J. Smith, H. Hendrick, P. Carayon, and M. Peruse. 1995. *Work Related Musculoskeletal Disorders (WRMSDs): A Reference Book for Prevention*. London, England: Taylor & Francis.

Haggerty, B., and P. Tarasewich. 2005. A new stylus-based method for text entry on small devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction*, Vol. 4, Las Vegas, July 22–27, 2005. Mahwah, NJ: Lawrence Erlbaum Associates.

HCII. 2005. *Proceedings of the 11th International Conference on Human-Computer Interaction*, July 23–27, 2005, Las Vegas. Mahwah, NJ: Lawrence Erlbaum Associates.

HCII. 2007. *Proceedings of the 12th International Conference on Human-Computer Interaction*, LNCS 4550-4566, July 22–27, 2007, Beijing, China. Berlin and Heidelberg: Springer.

HCII. 2009. *Proceedings of the 13th International Conference on Human-Computer Interaction*, LNCS 5610-5624, LNAI 5638-5639, July 19–24, San Diego, CA. Berlin and Heidelberg: Springer.

HCII. 2011. *Proceedings of the 14th International Conference on Human-Computer Interaction*, July 9–14, 2011, Orlando, FL. Berlin and Heidelberg: Springer.

Hendrick, H. 1986. Macroergonomics: A conceptual model for integrating human factors with organizational design. In *Human Factors in Organizational Design and Management*, ed. O. Brown and H. Hendrick, 467–77. Amsterdam, the Netherlands: Elsevier Science Publishers.

Hignett, S. 2007. Physical ergonomics in health care. In *Handbook of Human Factors and Ergonomics in Health Care and Patient Safety*, ed. P. Carayon, 309–21. Mahwah, NJ: Lawrence Erlbaum Associates.

Hinckley, K. 2008. Input Technologies and Techniques. In *The Human-Computer Interaction Handbook*, 2nd ed., ed. A. Sears and J. A. Jacko, 161–76. New York: Lawrence Erlbaum Associates, Taylor & Francis Group.

Hirose, M., and K. Hirota. 2005. PUI (Perceptual User Interface). In *Proceedings of the 11th International Conference on Human-Computer Interaction,* Volume 5, Las Vegas [Electronic publication]. Mahwah, NJ: Lawrence Erlbaum Associates.

Hultgren, G., and B. Knave. 1973. Contrast blinding and reflection disturbances in the office environment with display terminals. *Arbete Och Halsa*.

Lai, J., C.-M. Karat, and N. Yankelovich. 2008. Conversational speech interfaces and technologies. In *The Human-Computer Interaction Handbook*, 2nd ed., ed. A. Sears and J. A. Jacko, 381–91. New York: Lawrence Erlbaum Associates, Taylor & Francis Group.

Lawler, E. E. 1986. *High-Involvement Management*. San Francisco, CA: Jossey-Bass Publishers.

Mandal, A. C. 1982. The correct height of school furniture. *Hum Factors* 24(3):257–69.

Marcus, M., F. Gerr, C. Monteilh, D. J. Ortiz, E. Gentry, S. Cohen, A. Edwards, C. Ensor, and D. Kleinbaum. 2002. A prospective study of computer users: II. Potential risk factors for musculoskeletal symptoms and disorders. *Am J Ind Med* 41(4):236–49.

Myers, B. A., and J. O. Wobbrock. 2005. Text input to handheld devices for people with physical disabilities. In *Proceedings of the 11th International Conference on Human-Computer Interaction*, Vol. 4, Las Vegas, July 22–27, 2005. Mahwah, NJ: Lawrence Erlbaum Associates, electronic publication.

Nachemson, A., and G. Elfstrom. 1970. Intravital dynamic pressure measurements in lumbar discs. *Scand J Rehabil Med* (Suppl 1):1–38.

NAS. 1983. *Video Terminals, Work and Vision*. Washington, DC: National Academy Press.

Noro, K. 1991. Concepts, methods and people. In *Participatory Ergonomics*, ed. K. Noro and A. Imada, 3–29. London, England: Taylor & Francis.

OSHA. 2008. *Computer Workstations eTool*. Washington, DC: Occupational Safety and health Administration. www.osha.gov/SLTC/etools/computerworkstations/.

Ostberg, O. 1975. Health problems for operators working with CRT displays. *Int J Occup Health Saf* 6:24–52.

Robertson, M. 2007. Health and performance consequences of office ergonomic interventions among computer workers. In *Ergonomics and Health Aspects (LNCS 4566)*, ed. M. J. Dainoff, 135–43. Berlin and Heidelberg: Springer-Verlag.

Robertson, M., E. Huang, and N. Larson. 2009. Examining the effects of workstation design satisfaction. Computer usage, supervisory and co-worker support on perceived physical discomfort and psychosocial factors. In *Ergonomics and Health Aspects (LNCS 5624)*, ed. B.-T. Karsh, 88–94. Berlin and Heidelberg: Springer-Verlag.

Saito, S., B. Piccoli, M. J. Smith, M. Sotoyama, G. Sweitzer, M. B. G. Villanuela and R. Yoshitake. 2000. Ergonomics Guidelines for using notebook personal computers. *Industrial Health* 38:421–434.

Smith, M. J. 1984. Health issues in VDT work. In *Visual Display Terminals*, ed. J. Bennet, D. Case, J. Sandlin, and M. J. Smith, 193–228. Upper Saddle River, NJ: Prentice Hall.

Smith, M. J. 1987. Mental and physical strain at VDT workstations. *Behav Inf Technol* 6(3):243–55.

Smith, M. J. 1997. Psychosocial aspects of working with video display terminals (VDT's) and employee physical and mental health. *Ergonomics* 40(10):1002–15.

Smith, M. J., and P. Carayon. 1995. New technology, automation and work organization: Stress problems and improved technology implementation strategies. *Int J Hum Factors Manuf* 5:99–116.

Smith, M. J., P. Carayon, and W. Cohen. 2003. Design of computer workstations. In *The Human-Computer Interaction Handbook*, ed. J. Jacko and A. Sears, 384–95. Mahwah, NJ: Lawrence Erlbaum Associates.

Smith, M. J., P. Carayon, and W. J. Cohen. 2008. Design of computer workstations. In *The Human-Computer Interaction Handbook*, 2nd ed., ed. A. Sears and J. Jacko, 313–26. New York: Lawrence Erlbaum Associates, Taylor & Francis Group.

Smith, M. J., and W. J. Cohen. 1997. Design of Computer Terminal Workstations. In *Handbook of Human Factors and Ergonomics*, 2nd ed., ed. G. Salvendy, 1637–88. New York: John Wiley & Sons.

Smith, M. J., B. G. F. Cohen, L. Stammerjohn, and A. Happ. 1981. An investigation of health complaints and job stress in video display operations. *Hum Factors* 23(4):387–400.

Smith, M. J., and A. Derjani-Bayeh. 2003. Do ergonomic improvements increase computer workers' productivity? An intervention study in a call center. *Ergonomics* 46(1):3–18.

Smith, M. J., and P. C. Sainfort. 1989. A balance theory of job design for stress reduction. *Int J Ind Ergon* 4:67–79.

Smith, M. J., L. Stammerjohn, B. Cohen, and N. Lalich. 1980. Video display operator stress. In *Ergonomic Aspects of Visual Display Terminals*, ed. E. Grandjean and E. Vigliani, 201–10. London, England: Taylor & Francis.

Stead, W. W., and H. S. Lin. 2009. *Computational Technology for Effective Health Care: Immediate Steps and Strategic Directions*. Committee on Engaging the Computer Science Research Community in Health Care Informatics, National Research Council. Washington, DC: National Academy Press.

WorkSafeBC. 2009. *How to Make Your Computer Workstation Fit You*. Vancouver, BC: Worker's Compensation Board of British Columbia. www.WorkSafeBC.com/publications/health_and_Safety/by_topic/assets/pdf/comptr_wrkstn.pdf.

Part III

Designing Human–Computer Interactions

14 Visual Design Principles for Usable Interfaces

Everything Is Designed: Why We Should Think before Doing

Suzanne Watzman and Margaret Re

CONTENTS

Take a moment and visualize Las Vegas at night. What kind of image does this conjure up for you? Flashing lights from all directions, a hotel's lighting display designed to outdo its neighbor as well as that of the casino signage down the street. At first glance, everything is exciting, colorful, and beautiful. Now add a fireworks display to your picture. More color, more excitement. Where do you look first? There is so much going on; it is hard to see it all, but you don't want to miss a thing. Your head turns in all directions. You look there; then, out of the corner of your eye, you see something else. Look over there! Now the fireworks are at their peak, and the noise gets even louder. Any conversation with companions is impossible, yet it is also impossible to focus on any one thing for more than a split second. You are overwhelmed and overloaded. Everything is screaming for your attention. Can you manage to pay attention? For how long? Do you begin to shake your head in despair, and give up? Do you wish you were somewhere else—NOW?

14.1 MAKING THINGS EASIER TO USE AND UNDERSTAND: THINKING ABOUT THE USER'S EXPERIENCE

The previous description is unfortunately an accurate analogy of many users' experiences as they attempt to learn, work, play, and relax. New products, new services, and new technology with which you are unfamiliar can create confusion. Users of these new products, services, and technology are customers, electricians, grandparents, clerks, pilots, and students—you and me. And for most of us, it's a jungle out there! Las Vegas at night with fireworks, or monitors that are winking, blinking, distracting, disturbing, overwhelming—and, after a short time, visually deafening. Now add voices coming from boxes . . .! Although this may seem like an exaggeration, for many this situation is exactly their experience. User interface design focuses on designing flexible environments that have a positive impact on a user's ability to experience and interact with a product, whether that product is a mobile communication device, website, information kiosk, or appliance. It involves creating environments that include strong navigational devices that can be understood intuitively and used effortlessly. Designers have a responsibility to create user experiences that are simple and transparent. To do their job well, they must advocate on behalf of the user, ensuring that the interfaces they design are not just merely exercises in technology but that they truly assist and guide the user from task to task, enabling work to be done, and ultimately improving quality of life. When designers succeed, their products can be used effortlessly and are even pleasurable to use. Good design does not needlessly draw attention to itself. It just works. This is the role of good design.

14.2 DEFINING VISUAL DESIGN

The nautilus shell is an example of the synthesis between form and function found in nature (Figure 14.1). Its form is the result of evolution, which is both transparent and beautiful. The nautilus shell is a perfect analogy for design and the design process because it creates valuable user experiences and usable interfaces.

The word *design* functions as both a noun and a verb. Many people use it to refer to the outward appearance or style of a product. However, design also refers to a process—that of intentionally establishing a plan or system by which a task can be accomplished or a goal reached. It includes tangible and intangible systems in which objects or processes are coherently organized to include the environments in which these objects or processes function. Design affects all people in every aspect of what they do. Good design performs for people. It is concerned with economics and the transmission of ideas. The challenge presented to a design team is to plan a prototype with a clear purpose that is easy to use, meets user needs, addresses commercial considerations, and can be mass-produced. Its visual form, whether two- or three-dimensional, digital or analog, logically explains its purpose and efficiently leads the user through its function. Design is not a series of subjective choices based on personal preference, at best a cosmetic afterthought considered if time and

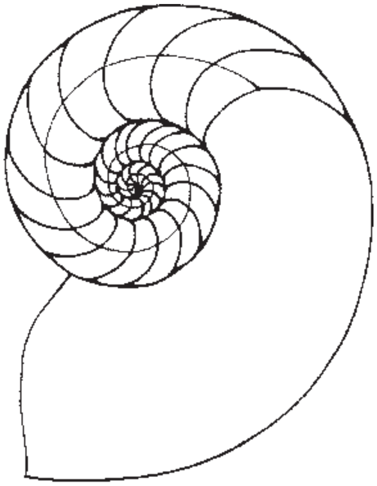


FIGURE 14.1 Nautilus shell.

money are leftover. Good design is the tangible representation of product goals. An iterative and interactive process that requires active learning, design unifies a wide range of disciplines. Good design is a significant activity that reveals multiple solutions to each problem. Design equally values different ways of thinking. It allows people with a variety of skills and learning abilities to work cooperatively to bring insights and expertise to problems and opportunities in order to better develop new and innovative solutions. Problems can be analyzed using a multitude of viewpoints and methods. Writing, drawing, statistical analysis, graphing, discussion, interviewing, personal observation, model-making, and diagramming are all legitimate methods for examination as the physical, social, and cultural contexts of possible answers are considered (Davis et al.).

14.3 DESIGN PROCESS

Design as a Catalyst for Learning, a publication funded in part by the National Endowment for the Arts, argued that effective design that responds to human problems uses the following steps (Figure 14.2):

- *Problem identification and definition*: A need or problem is identified, researched, and defined.
- *Gathering and analyzing information*: The focus is on learning what is not known. Assumptions are questioned. Wide and broad research is used to locate information and generate ideas.
- *Determining performance criteria for a successful solution*: Research continues as imagery is selected. Rules are declared and what is known is specified.
- *Generating alternative solutions and building prototypes*: Multiple solutions are generated. A variety of methods for analysis, such as drawing, interviewing, modeling or evaluating statistics, are used.

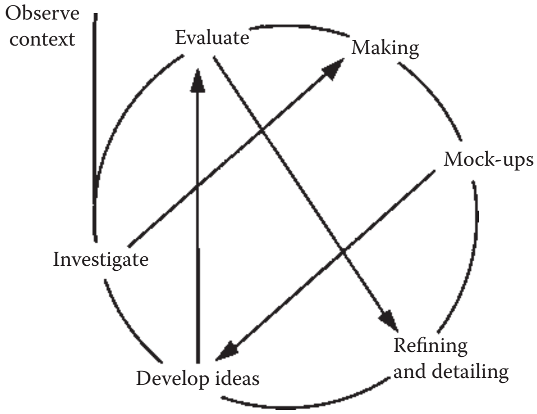


FIGURE 14.2 Interacting design loop. The interacting design loop developed by Richard Kimbell, founder of the Technology Education Research Unit at Goldsmiths College, and presented in *Design as a Catalyst for Learning*, captures the divergent, iterative and cyclical nature of the design process.

- *Implementing choices*: Project content, scope, and intent are formally established. Initial possibilities are represented and presented as prototypes.
- *Evaluating outcomes*: Prototypes are assessed, tested, evaluated, and judged. The knowledge gained is incorporated into further studies and refinements.
- *Production*: A prototype, which is a synthesis of the initial solutions made using this process, and specifications are released for making multiples to a manufacturer.

14.4 ROLE OF THE DESIGNER

Visual design decisions are based on project goals, user perspective, and informed decision making. While many aspects of design are quantifiable, there are visual principles that are less measurable but equally important. Even though the necessary skills to become visually literate and make competent design decisions can be learned, design involves a highly specialized knowledge base. A unique combination of creativity and skill differentiates and makes one design more attractive and desirable than another. Both education and talent are necessary to apply the principles required to present information in its most accessible, useful, and pleasing form. The role of the designer in the development of interfaces for interactive products is to understand the product goals and ensure that information is approachable, useful, and desirable. In an environment in which the interface is the only tangible representation of a product and user perception determines product success, appropriate information presentation and visual design are key. Designers understand visual principles in context, and know how to apply them appropriately to create innovative, functional and aesthetically pleasing solutions.

14.5 PROCESS OF GOOD DESIGN—HOW DO WE GET THERE FROM HERE?

Interface designers are responsible for defining what the experience will be like when a product is used. While print media dictates that users encounter content in a largely predetermined sequence, an interface offers the user greater flexibility over how content can be accessed based on users' needs and wants. A successful interface can be easily navigated. Interface designers define, decide, and then create the experience for users, so that an experience with a product is useful, meaningful, even pleasant and empowering. The designer must maintain an attitude of unbiased discovery and empathy for the user. The designer must develop clearly defined goals in order to create a good design that includes an evaluation process that supports and enhances these goals, and includes the flexibility to respond to changes as the process continues and products evolve.

14.6 INFORMATION-DESIGN PROCESS IS AN INFORMED DESIGN PROCESS

An information-design process (IDP) is a method of visually structuring and organizing information to develop effective communication. Information design is not superficial or decorative, but is rather a merging of functional, performance-based requirements with the most appropriate form to present these requirements. A thoughtful, well-designed solution will

- *Motivate users*: It psychologically entices an audience, convincing members that information and tasks can be successfully handled.
- *Increase ease of use and accessibility*: The effort needed to comprehend information is decreased. A clear path that aids in skimming and referencing text and gives easy access is provided.
- *Increase the accuracy and retention of the information*: Users learn and retain information better when it is visually mapped and structured in obvious and intuitive ways.
- *Focus on the needs of its users*: Multiple audiences have different requirements and styles of learning. Solutions should be developed that provide alternative means of accessing information for different types of users. An information-design approach is part of a process that incorporates research, design, testing, and training to produce useful, cost-effective solutions.

14.6.1 Phase 1: Audit

The goal of the audit is to create a blueprint for the project, much like architectural drawings are developed before constructing a building (Figure 14.3). The audit process begins by asking and answering a number of questions and acknowledging ongoing change and an ever-increasing palette of products and services. Questions are asked throughout the entire product lifecycle, since the answers/design solutions reflect the user/use environment and affect the ongoing usefulness and value of the product. To create an eloquent design, continually ask and answer the following questions:

14.6.1.1 Audit Questions A

- Who are the product users?
- How will this product be used?
- When will this product be used?
- Why will this product be used?
- Where will this product be used?
- How will the process evolve to support this product as it evolves?

After the first set of questions are asked and answered, a second set of questions must be asked and answered:

14.6.1.2 Audit Questions B

- What is the most efficient, effective way for a user to accomplish a set of tasks and move on to the next set of tasks?
- How can the information required for product ease of use be presented most efficiently and effectively?
- How can the design of this product be done to support ease of use and transition from task to task as a seamless, transparent, and even pleasurable experience?
- What are the technical and organizational limits and constraints?

An audit focuses on discovery. Many disciplines and organizational resources must be consulted. Change is a given, since designers begin with assumptions and don't know all they need to know yet. The answers to their questions and their analysis in the context of organizational objectives provide the basis for the audit report, which serves as the guide in design development. The audit report can be as simple as a two-page list or as complex as a comprehensive hundred-page report. Since the goal is discovery, it includes every aspect of the organization concerned with the product-development

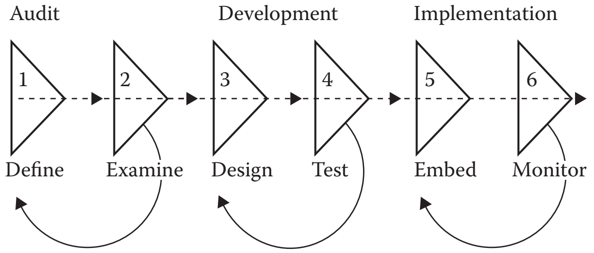


FIGURE 14.3 Information design process (IDP). IDP is phased to insure user and organizational needs are met. It is ongoing and iterative, throughout the lifecycle of a product. Any change can trigger a recycling of this process, to insure solutions remain appropriate and useful.

cycle: project management, usability engineering, technical development, user support and documentation, visual communication and design, and content management. With these goals, the result is unbiased, accurate, comprehensive information that serves as the basis for design.

14.6.2 Phase 2: Design Development

The design-development phase uses the audit report as a guideline. This is an ongoing, iterative process with each iteration incorporating user test results to make the product appropriate to the particular set of needs. In reality, the length of this process is often defined and limited by real-world deadlines such as product release dates. The design-development phase includes design and testing. The designer or design team creates a number of solutions based on results and objectives determined by the audit report as well as other project specifics. Initially, design ideas should be very broad, incorporating many ideas and options no matter how unrealistic or unusual. As ideas are tested, user feedback incorporated, and other parameters defined, solutions naturally become more defined. Surviving design ideas are then based on solid information derived from user feedback, providing a strong basis for final design decisions. In the beginning, the focus is on high-level concepts and navigation. How will the product work? What will it feel like to use? As initial concepts are refined, design details become more specific. When the conceptual model and organizational framework are approved, the design of the look or product package begins. By the end of this phase, a prototype design to be carried out in implementation and monitoring is tested, approved, and specified.

14.6.3 Phase 3: Implementation and Monitoring

The implementation phase focuses on delivering what has been defined, designed, and documented in the preceding phases. It is the final part of a *holistic* process that defines everything necessary to make a product succeed on an ongoing basis. This includes not only the implementation of the design within the technology, but also any additional support such as the creation of training materials and other reinforcements that enhance use and productivity. Continuous monitoring is key to sustained success, because a successful product responds to evolving technology and user needs. This last phase is mostly consultative and ongoing throughout the product lifecycle in order to ensure that changes such as new technology and product developments are reflected in the product itself. These may in fact trigger another audit/design/testing cycle, although usually less extensive than the initial process. Though the implementation phase is called "the last phase," it reveals the evolutionary process of design and development. The goal of ongoing monitoring of solutions is to be aware of changes in user needs, technology, and competition that impact user acceptance and satisfaction. Changes here often result in the need to reevaluate and redesign to incorporate this new knowledge gained.

14.7 VISUAL DESIGN PRINCIPLES

Interaction design bridges many worlds: that of visual design, information presentation, and usability with aesthetics. Donis (1973a), in *A Primer of Visual Literacy*, argued that art and its meaning have dramatically changed in contemporary times from one that involved a concern with function to one that views the process of creating art as that of making emotional maps that spring from the province of the intuitive and subjective. This argument extends to design. To someone unskilled in creating effective communications, visual design is often understood as personal preference limited to style or appearance. However, any form of effective design is a result of rigorous study, a concern for organization and usability combined with knowledge of the basic design principles of harmony, balance, and simplicity. Visual design is in fact a form of literacy.

14.8 UNIVERSAL PRINCIPLES OF VISUAL COMMUNICATION AND ORGANIZATION

The principles of harmony, balance, and simplicity are related yet distinct in meaning and application. *Harmony* is the grouping of related parts, so that all elements combine logically to make a unified whole. In interface design, as with other categories of design, this is achieved when all design elements work in unity. Transitions from place to place are effortless and the techniques used to achieve this harmony are unnoticed by the user. Visual harmony achieves the same goal as musical harmony in which notes combine to create a chord. The golden section, also known as the "golden mean" or "golden rectangle," is one of the most widely used methods for creating harmony. Architects, artists, musicians, mathematicians, and designers have used the golden section extensively for centuries to create proportional relationships (Figure 14.4).

Balance offers equilibrium or rest. Donis stated that equilibrium is the strongest visual reference (Donis 1973b). It provides the equivalent of a center of gravity that grounds the page. Without balance, the page collapses, all elements are seen as dispersed, and content is lost. Balance requires continual modification from page to page because while each page is part of a greater system, elements can vary and all have visual weight. In the same way that a clown balancing on a ball while juggling objects of different weights must continually make adjustments for actions that are occurring, visual balance requires the same concerns and adjustments as in the physical world. Regardless of how a design is organized, it must achieve stability and unity in order for a user to feel comfortable with the solution. Balance can be achieved a number of ways. One obvious method uses *symmetry*, such as found on a page with text and image aligned on a centered axis. Deceptively simple, symmetry form is often considered easy to make; however, unless handled carefully a symmetrical composition can be predictable, boring, and static. *Asymmetry* employs nonaxial balance and uses contrast between elements such as weight,

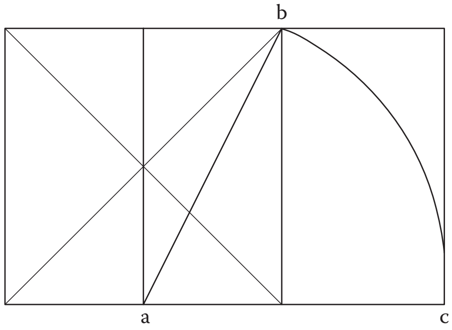


FIGURE 14.4 The golden rectangle. Divide a square to find the center point (a) from which length (ab) is found. From point (a) the length (ab) is swung as an arc to point (c) to create a rectangle that uses the proportions of the golden section.

form, and color to create visual tension and drama. Both are valid approaches and require skill and knowledge of complex visual interaction to achieve.

Simplicity is the embodiment of clarity, elegance, and economy. A solution that offers simplicity is unambiguous and easily understood. It offers clarity working effortlessly devoid of unnecessary decoration. It appears deceivingly easy, accessible, and approachable, even though it may be conceptually rich. Simplicity involves distillation—every element is indispensable, if an element is removed, the composition falls apart. Achieving simplicity is no easy task. Two guidelines for creating simple design solutions are (1) "Less is more!" (attributed to Mies van der Rohe) and (2) "When in doubt, leave it out!" (Anonymous). The most refined design is direct and includes only essential elements. Removing any element breaks the composition rendering it unintelligible or radically different.

14.9 VISUAL DESIGN TOOLS AND TECHNIQUES

New technologies are rapidly being created that extend past the simple automation of tasks and communication; they are revolutionizing processes and the resulting products. Before the revolution brought about by electronic publishing technology, many disciplines such as writing, editing, design, publishing, and programming were discrete units that handled a defined step within a larger process. Today's new publishing environments encourage the possibility of a *renaissance* publisher—a person who can create, design, publish, and distribute. Yet the process used to arrive at successful solutions is very complex and extends past technical knowledge to include a mastery of visual and verbal language in order to build effective communication. Focus must be placed on the factors that determine success with constant evaluation and adjustment of these factors in light of new developments.

14.9.1 Five Criteria for Good Design

Before any work begins, all participants in the process should have a clear understanding of the criteria for good design. The following questions are guidelines for evaluation of design solutions before, during, and after the process to ensure that all solutions remain valid as products, technology, and user needs evolve.

- ***Is it appropriate?*** Is the solution appropriate for the particular audience, environment, technology, and/or culture?
- ***Is it durable?*** Will the solution be useful over time? Can it be refined and transitioned as the product evolves and is redefined?
- ***Is it verifiable?*** Has the design been tested by typical users in the environment that it will be used in? Has feedback been properly evaluated and used to improve the product?
- ***Does it have impact?*** Does the design solution not only solve the problem, but also impact the look and feel, so that the user finds the product experience comfortable, useful, and desirable?
- ***Is it cost effective?*** Can the solution be implemented and maintained? Are individuals with the necessary skills and understanding to create, refine, and maintain the design available throughout the product's life? The cost of any design begins with the audit and design phases, but continues after the implementation phase to insure that it remains advantageous and cost-effective. The hard and soft costs of delivering the solution plus ongoing maintenance add up to the real design costs.

14.9.2 Visual Design Principles at Work

The following sections outline the disciplines and principles used to create quality design solutions. Each topic is worthy of extended study, because there is much to understand when evaluating how to effectively present information. As the design process evolves, insights and information are discovered that impact a solution. It is optimistic to base solutions on an initial exercise because the very nature of process means discovering what is unknown yet critical. Therefore, all members involved in the design process must remain open and ready to incorporate new information, which may change or delay results, but more accurately reflect user needs.

For example, if a new feature is developed that changes a product's target audience from mid-level managerial to executive users, most methods for critical interaction and content delivery should be reconsidered. Executives have less time and need different information. The result might be a simpler interface with streamlined content that uses a larger typeface and a more conservative visual language. The most important principle to remember when thinking about design is that there are no rules, only guidelines. Everything is context sensitive. Always consider and respect the users.

14.9.3 Typography

In *The Elements of Typographic Style*, the poet and typographer Bringhurst (2005) described typography as frozen language. *Typography* is the visual representation of spoken and unspoken thought that allows an idea to be shared across time and distance independent of its creator. A functional and expressive art that shares many of the same concerns as writing and editing, typography involves organizing text so that its meaning is communicated according to an author's intent. In design, a literacy that provides an understanding of typography and how text can be structured in space is as important as a literacy that understands how to structure grammar in order to explicate content.

Typography is made from type, individual characters organized by visual characteristics into typefaces. Type is the smallest definable part of a design, much like a pixel is to a screen display. Felici (2003), who has worked through evolutions in typesetting technologies in *The Complete Manual of Typography*, defined a font as the electronic file that contains the programming code needed to make the characters found in a typeface. Historically, a typeface consists of all the individual characters or glyphs at a given size: letterforms, punctuation, numbers, mathematical symbols, diacritical marks, and other accessory characters needed to fully compose a text. This definition serves as a reminder to read a text carefully and consider all needs before selecting a typeface and developing a presentation form (Figure 14.5).

Effective typography is rational. It is concerned with clarity and comprehension; the ease in which characters and word shapes are recognized in reading environments and the ease in which they can be used. It extends past the shapes of individual letters and their potential combinations to include the relationships found between word and interword shapes, functional groupings that ultimately progress into issues of type weight, slope, width and scale, characteristics that act as interpretative devices in order to create influential and persuasive form. If the principles of good typography can be understood and applied, then these same principles can be extended to more complex issues that follow such as page and product design. Typographic choice affects legibility and readability, the ability to easily see and understand what is on the page, in all media. Tracy (1986), in *Letters of Credit: A View of Type Design*, offered the most useful definitions for legibility and readability. *Legibility*, the speed at which letters and the words built from them can be recognized, refers to perception. *Readability*, the facility and ease with which text can be read, refers to comprehension. Legibility and readability are related. Regardless of media, legibility and readability are determined by variables such as point size, letter pairing, word spacing, line length and leading, resolution, color, and organizational strategies such as text clustering. Together, legibility and readability comprise typography's functional aspects. Good typography, like good design, is invisible to the user—it just works (Figure 14.6).

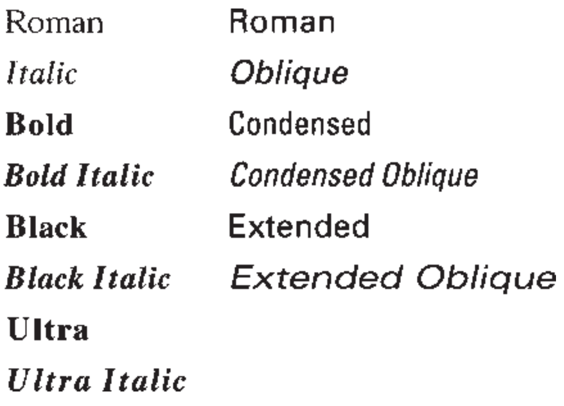


FIGURE 14.5 Type family. A type family is built around four core members: roman, bold, italic, and bold italic. Additional members may include typefaces whose weight and width are variants of the core group. A family can also contain expert sets that offer additional or alternate characters such as small caps, fractions, and non-aligning numbers.

Selecting an appropriate typeface for a specific purpose and context requires experience and understanding (Figure 14.7). With thousands of typefaces to choose from and numerous ways to manipulate them, finding the typeface best suited for an audience is not easy. With its lack of control, multiple media, and varied viewing contexts, the current publishing environment makes this a complex task.

Typeface choice impacts whether and how a communication is read. Distinct typefaces and typographic styles create environments that influence a user's perception of text. The physical nature of the presentation itself helps determine content and acceptance. A typeface with extremely thick and thin strokes may appear sophisticated and readable in print but may look naïve and render text unreadable in a digital environment. Typefaces are frequently designed to solve issues of legibility and readability created by a technology. A typeface made for online use can increase page legibility, as well as the overall perception of approachability, quality of an interface, and ultimately product acceptance (Figures 14.8 through 14.10).

An informed selection can make reading enjoyable and effortless rather than frustrating and fatiguing. Though typography might seem to be an insignificant issue to a non-designer, it affects overall usability. A clear understanding of the concepts and principles that affect legibility and readability is crucial to determining effective typography.

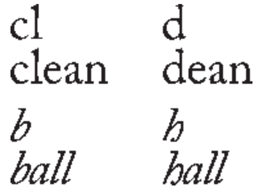


FIGURE 14.6 Legibility and readability. The letters, letter pairs, and words shown above are examples of what can happen if the designer is not sensitive to issues of legibility and readability.

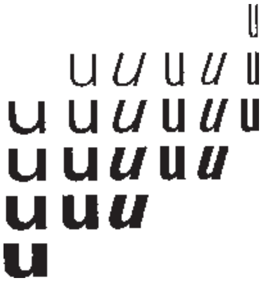


FIGURE 14.7 Univers “U”. Univers, a type family designed by Adrian Frutiger and released for commercial use in 1954, is composed of twenty-one fonts that together offer a wide range of weights, widths, and slopes that allows a text to be organized so that its form is visually coherent and easily read.

Bell Centennial, 6 point

Address
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890 ([.,;:"-"/—)

Name & Number
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890 ([.,;:"-"/—)

Bold Listing
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890 ([.,;:"-"/—)

Sub-caption
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890 ([.,;:"-"/—)

FIGURE 14.8 Bell centennial: Technology-specific typefaces. AT&T commissioned Bell Centennial, a typeface designed at a very small size, for telephone directory use, in order to solve an industrial problem created by changing typesetting and printing technologies. The resulting type family designed for maximum legibility, readability, and spatial efficiency provided the user with a clear information hierarchy. It reduced paper use and directory assistance calls. Here, Bell Centennial is shown at six point, the size at which it was intended to function.

acefgsv
0123456789

acefgsv
0123456789

FIGURE 14.9 Bell centennial: Technology-specific typefaces. Bell Centennial’s forms were opened to increase legibility and readability. Curved strokes were straightened and the horizontal and vertical juncture were notched so that they did not clog with ink when printed. Select letterforms shown at 32 point for Bell Centennial (bottom) and Helvetica (top) illustrate this.

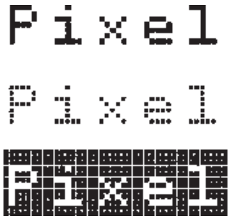


FIGURE 14.10 Pixel font. Pixel fonts are screen fonts designed specifically for use with or as on-screen navigational elements. Their block-like forms are the result of working with the actual pixels themselves.

14.10 HOW THE HUMAN EYE SEES, AND THEN READS

Spencer (1969) in *The Visible Word*, a publication with an objective of introducing and uniting those who research legibility with those who work with typography, presented that the eye uses both outline word shapes and their internal patterns to move along a text line and steps and jumps as it groups text to form comprehensible phrases of information. This motion of the eye during reading is known as “saccadic movement.” Sometimes during this process, the eye regresses and returns to what has been read. Optimal typography allows for fewer backward movements. Spencer argues that possessing a mastery of reading mechanics is important to a mastery of content. Typeface selection directly affects this skill, making it easier or more difficult for the eye to group, read, and understand information (Figures 14.11 through 14.13).

14.11 TYPEFACE SIZE AND SELECTION

Type size is given in points, a measuring system unique to typography. In digital typesetting systems, a point equals 1/72 of an inch. Type size cannot be determined by physically measuring a letterform because when type existed solely in metal, the technology in which it was first used, size was measured by the height of the metal block on which a letter sat. This is one of the reasons why the same letter repeatedly set in differing typefaces at the same point size appears dissimilar when compared. Lowercase letters set in the same point size with differing typefaces can appear larger or smaller in comparison to each other due to variations in x-height. Other variables such as stroke weight, letter width, and ascender and descender length influence size perception and help make some typefaces more or less readable and legible than others. Type size is also dependent on the resolution offered by output and viewing devices, color usage, context, and other design issues. It is crucial to understand not only the characteristics of a typeface but also usage context and application environment when selecting a typeface.

14.11.1 Serif and Sans Serif

Serif and sans serifs are general categories used for classifying type. *Serif* refers to a typeface with serifs, the short strokes that project off the end of letter strokes, as opposed to *sans serif*, a typeface without serifs. It is debatable whether serif typefaces, conventionally used for setting text in print, are more readable than sans serif typefaces, which have an even stroke weight and more open counters that have proven useful for setting text for on-screen reading. And, while sans serif is considered easier to read on screen, serifs can be made equally legible if the appropriate typeface, size, and color is specified. Some designers think that, in print, serifs aid in character recognition and readability; they help differentiate individual letters creating horizontal lines for the eye to follow. This has not been proven conclusively. Other designers hold the view that "we read best what we read most." It's likely that this discussion will continue. Recent technological developments that subdivide a pixel into red, green, and blue elements on LCD screens have resulted in new technologies that create a better immersive environment for online reading. This, in turn, will spark new explorations in typographic form and its presentation.

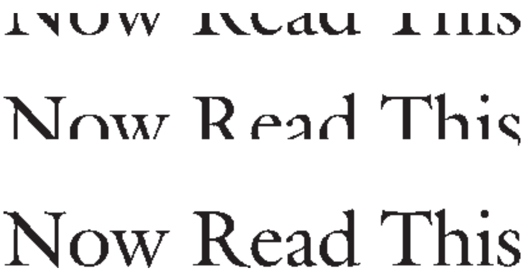


FIGURE 14.11 Now Read This. The phrase, Now Read This, is shown in full, cropped at the bottom and from the top, suggesting the importance of outline word shape and internal pattern.

14.11.2 Families of Type

Many design students first learn to classify typefaces into five chronological families or organizational groupings popularized by Craig et al. (2006) in *Designing with Type*: (1) old style, (2) transitional, (3) modern, (4) Egyptian, and (5) contemporary. Classifying typefaces into these families makes it

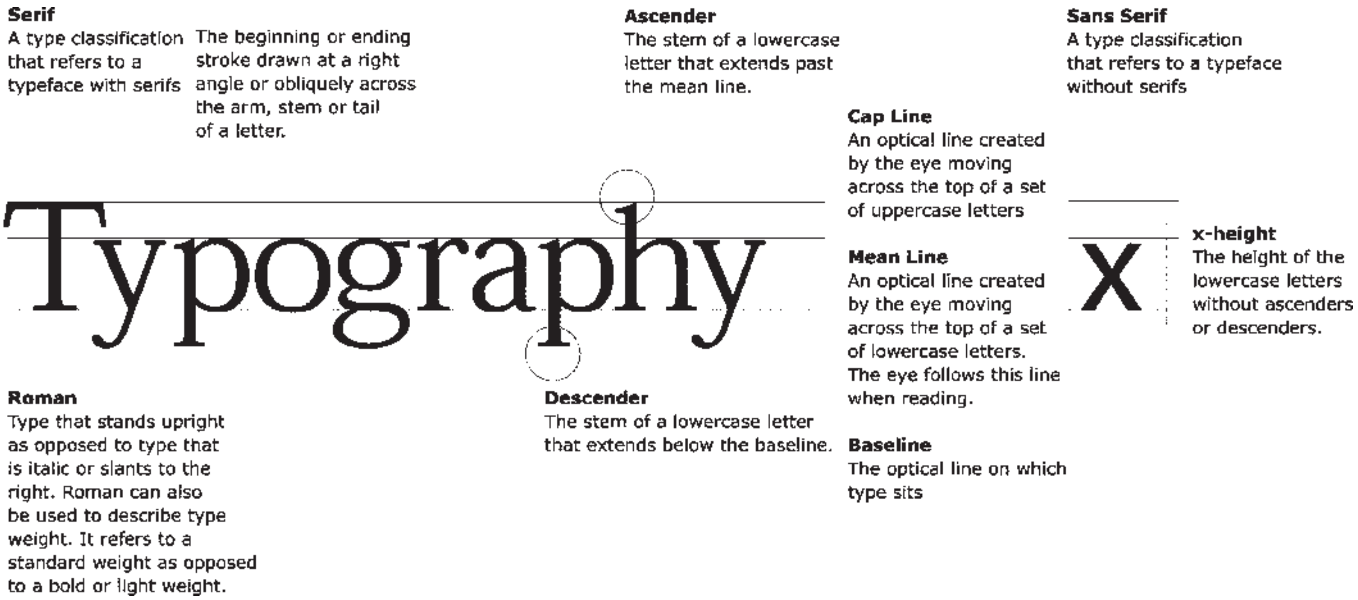

Serif
A type classification that refers to a typeface with serifs
The beginning or ending stroke drawn at a right angle or obliquely across the arm, stem or tail of a letter.
Ascender
The stem of a lowercase letter that extends past the mean line.
Sans Serif
A type classification that refers to a typeface without serifs
Cap Line
An optical line created by the eye moving across the top of a set of uppercase letters
Typography
x
x-height
The height of the lowercase letters without ascenders or descenders.
Mean Line
An optical line created by the eye moving across the top of a set of lowercase letters. The eye follows this line when reading.
Roman
Type that stands upright as opposed to type that is italic or slants to the right. Roman can also be used to describe type weight. It refers to a standard weight as opposed to a bold or light weight.
Descender
The stem of a lowercase letter that extends below the baseline.
Baseline
The optical line on which type sits

FIGURE 14.12 Anatomy of a letter. The typographic terms defined and illustrated above are used by designers in discussing principles that affect the legibility of type and overall quality of the communication.

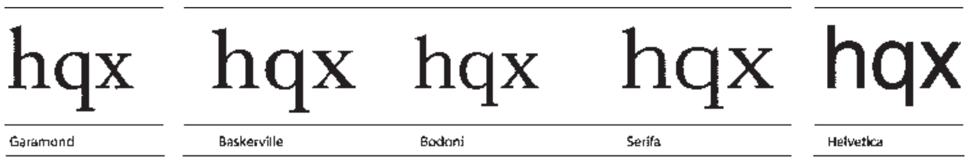


FIGURE 14.13 Hqx illustration. The x-height of a typeface (based on the actual height of a lower-case x) is a key characteristic when deciding the visual size of a typeface, particularly when readability is critical. While the above typefaces are the same point size, some seem larger (e.g., Helvetica) and easier to read than others (e.g., Serifa).

easier to understand the differences and similarities in structure and fitness. Like anything else, type design does not happen in isolation. These categories, as with those of many other classification systems, are formed around historical junctures in which the overall design of letterforms shifted dramatically in response to technological, political, cultural, aesthetic, and economic concerns. Typefaces within each set have passed through the tenures of metal, photocomposition, and have been adapted for digital technologies. Many of the problems solved through these older forms have proven inspirational to contemporary type designers who, as their predecessors did, continue to explore new forms for new purposes.

Verdana, a sans serif screen font designed by Matthew Carter, whose roots lie in the Industrial Revolution, was commissioned by Microsoft and released in 1996. Its form, which uses a visually even stroke with wide counters, helped signal a new software release. Verdana's members consist of Roman, italic, bold, and bold italic. It has two peers that use non-Latin alphabets: Verdana Greek and Verdana Cyrillic. Verdana is related to Tahoma, a condensed variation of Verdana designed for use in situations that require more information to fit in less space—such as with dialog boxes and menus; and Nina, a spatially efficient sans serif designed for situations that require more information to fit in even less space—such as with small handheld devices. Berry (2004), who writes and consults extensively on typographic matters, said in *Now Read This* that Meiryo, Verdana's daughter, evolved partially as a response to technologies that enhance photometric resolution permitting more complex writing systems such as scripts to be optimally read on screen, and a demand for a Japanese type that weaves the character sets of Kanji, Kana, Katakana and Romanji together, fashioning a favorable reading environment for screen and print. Meiryo, too, has Greek and Cyrillic companions. That one family has siblings and offspring, manufactured around a variety of alphabets and writing systems, speaks in part to the economic and political concerns of an international corporation that must respond to the demands of different cultural markets as much as it does to the need for multilingual communication (Figure 14.14).

14.12 VARIATIONS IN LETTERFORMS

14.12.1 Variations in Stress

Early type designers mimicked scribal letterforms because they knew and understood these forms. Old-style typefaces have a diagonal stress, a backwards slant—a visual remnant of the pen—that distributes weight through the thin part of the letterform. Over time, the stress traveled several degrees to the right as seen in transitional typefaces. With modern typefaces, the stress becomes rigidly vertical. Egyptians such as Century Schoolbook have a slight diagonal stress; other Egyptians with a stronger slab serif such as Serifa have no stress. Contemporary typefaces have no noticeable stress.

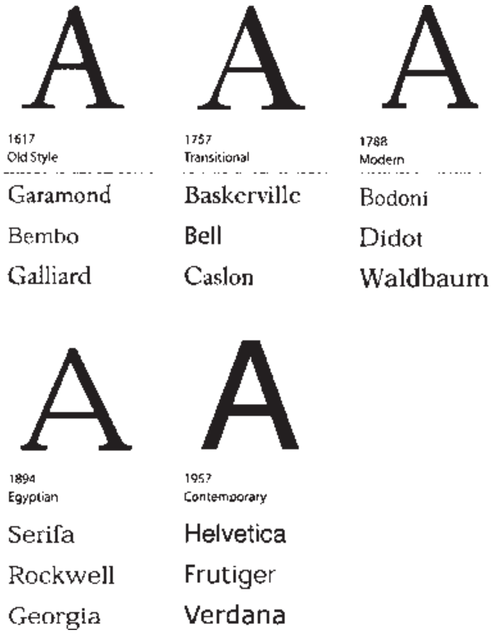


FIGURE 14.14 Type classification. These five A's are representative of typographic style from the 1600s to present day, reflecting changes in tools, fashion and current events. Ultimately, choice of output media should determine typeface selection, given details such as stress, the thick and thin parts of letterforms, negative space, viewing environment, output resolution, and so on.

14.12.2 Variation in Thick and Thin

The degree of contrast between the thick and thin strokes of the letters can vary. Old style typefaces have little contrast in strokes. This contrast increases in transitional faces. Extreme stroke contrast is a dominant characteristic of modern typefaces. Egyptian typefaces return to less contrast. Contemporary typefaces have no perceptible thick and thin strokes.

14.12.3 Variations in Serifs

Serifs differ in weight and bracket, which is the shape created by the serif joining the vertical stroke of the letter. Old-style typefaces have heavy concave serifs with thickset brackets. The meticulous serifs of modern typefaces are refined and thin and without brackets. Many Egyptians have heavy, straight serifs with little or no bracket. Sans serifs are considered contemporary typefaces (Figure 14.15).

14.13 TYPOGRAPHIC GUIDELINES

14.13.1 Combining Typefaces

Sans serif and serif typefaces can be effectively combined if changes are limited to prevent visual chaos. The key is to ensure that the result respects the content and reinforces the information hierarchy and overall design goals. When combining typefaces, decide whether harmony or contrast is important. Generally, do not use more than two different

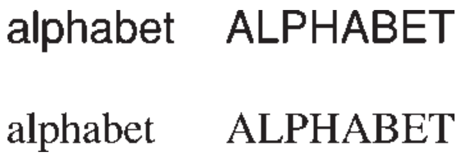


FIGURE 14.15 Serif versus sans serif. The serif typeface Century above versus the san serif typeface Univers, below. Understanding a typeface's physical characteristics and how it performs in different environments is important. Set a paragraph of text in both a serif and sans serif with exactly the same line length, size and spacing and compare the differences on screen and on paper.

type families in a document. Remember, at a minimum, a type family offers a Roman, italic, bold, and bold italic. Consider the pattern and texture that the x-heights and stroke weights weave when combined. Excellent typography does not impede the user and the information. Too many typefaces jar and confuse the reader, create visual intrusions, and slow the reading process.

14.13.2 Contrast in Weight (Boldness)

Combining two classic typefaces with a strong differential factor such as Helvetica Extra Bold with Times New Roman can add useful contrast. Be wary of combining intricate typefaces such as Gill Sans Bold and Souvenir, which have structures that may not create compatible reading environments. Too much contrast and visual complexity can be detrimental.

14.13.3 Output Device and Viewing Environment

The quality of publishing technologies and viewing environment vary greatly—laser printer versus video versus electronic media, and so on. In choosing a typeface, its style, size, spacing, and leading, think about the final output medium, and examine this technology's effect on legibility. Low-quality monitors and poor lighting have a major impact: serifs sometimes disappear, letters in small bold type fill in and colored type may disappear altogether.

14.13.4 Letter Spacing and Word Spacing

While the spaces within and between letterforms and words are determined by a type designer in order to set a rhythm that reads well, this spacing can be altered or kerned. However, be careful! When letter spacing is too tight, the letters are hard to distinguish from each other and legibility decreases. When letter spacing is too wide, letter groups are not easily recognized. Spencer argues that optimal letter spacing is inconspicuous, the user can read quickly and easily and understand content. Tight word spacing makes distinguishing individual words difficult. When word spacing is too wide, word groups fall apart. When there is greater space between words than there is between lines, the reader's eye naturally falls to the closest word, which may be below instead of across the line. This frequently occurs with low-resolution or low-cost products.

14.13.5 Line Spacing/Leading

Leading is the distance measured in points between the baseline of one line of text and the baseline of the text line below it. Ascender and descender length influences how closely lines of type can be stacked. The space between lines of text, or leading, should increase in relation to type size. This adjustment is visual not mathematical. Overall legibility may be improved by increasing the leading in relation to column width.

14.13.6 Line Length/Column Width

The correct line length is just long enough for the eye to sweep across without losing its place and easily drop down to continue reading the following lines. A good rule of thumb is that a line of average length contains between 39 and 52 characters.

14.13.7 Justified versus Ragged Right (Flush Left)

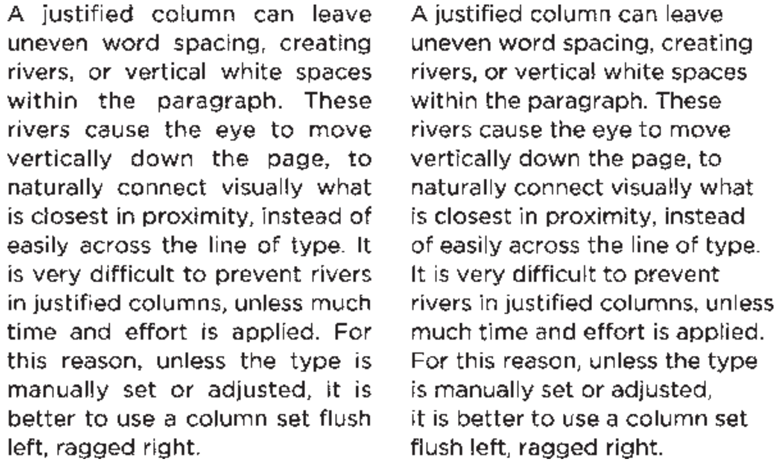
A justified text column can leave uneven word spacing, creating rivers, or vertical white spaces, within the paragraph. Rivers cause the eye to move vertically down the page, naturally connecting with what is closest in proximity, instead of moving easily across the line. It is very difficult to prevent rivers in justified text columns without spending considerable effort. Unless the type is manually set or adjusted, which is a time-consuming activity, it's better to use type that is set flush left, ragged right (Figure 14.16).

14.13.8 Highlighting with Type

Content can be highlighted by modifying type weight, slope, or case. Weight can be shifted from Roman to bold or extra bold. Slope can be altered from Roman to italic. Be mindful that italics are appropriate for short phrases and not long text passages. The italic appears lighter and smaller on the page when compared with its companion Roman and its complex forms are more difficult to read. Case can change from upper and lowercase to all capitals or small caps. Using all caps for extended text passages impedes readability since word outlines are rectangular and harder for the eye to differentiate. Limited shape and size cues are available to help differentiate between letters, words, and sentences to create meaning. Use only one highlighting technique for emphasis.

14.13.9 Decorative Typefaces

Decorative typefaces are of limited use for body text, because their irregular design lessens legibility and should be used in headlines with caution. Because they are essentially typographic fashion statements, decorative typefaces can either reinforce or distract from the overall message or brand of a particular product or organization.



A justified column can leave uneven word spacing, creating rivers, or vertical white spaces within the paragraph. These rivers cause the eye to move vertically down the page, to naturally connect visually what is closest in proximity, instead of easily across the line of type. It is very difficult to prevent rivers in justified columns, unless much time and effort is applied. For this reason, unless the type is manually set or adjusted, it is better to use a column set flush left, ragged right.

A justified column can leave
uneven word spacing, creating
rivers, or vertical white spaces
within the paragraph. These
rivers cause the eye to move
vertically down the page, to
naturally connect visually what
is closest in proximity, instead
of easily across the line of type.
It is very difficult to prevent
rivers in justified columns, unless
much time and effort is applied.
For this reason, unless the type
is manually set or adjusted,
it is better to use a column set
flush left, ragged right.

FIGURE 14.16 With current technology, the difference between a justified text column and ragged right text column can make a huge difference in readability. In a poorly justified column, spaces within a justified line connect vertically down the page, distracting the eye from easily reading across a line of text.

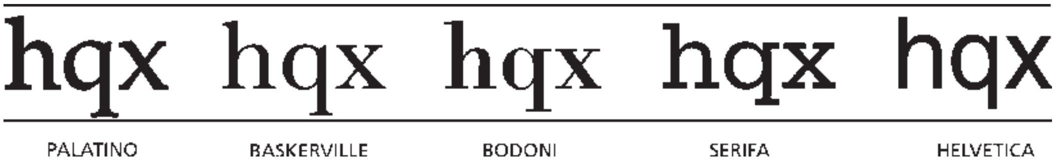


FIGURE 14.17 Black on white versus white on black. The x-height of a typeface (based on the actual height of a lowercase x) is a key characteristic when deciding the visual size of a typeface, particularly when readability is critical. While the above typefaces are the same point size, some seem larger (e.g., Helvetica) and easier to read than others (e.g., Serifa). Letterforms often appear as black/dark shapes on white/light backgrounds. The eye also reads the reverse, or negative shape around a letterform, which can create a shape that visually distracts and makes text difficult to read. Try setting a paragraph of black type on a white background, then set the same exact paragraph with white type on a black background. You can also try this same exercise with a dark color on a light background, and then try the reverse. You will notice that the greater the contrast (e.g. white type on black) the harder it is to read in large amounts.

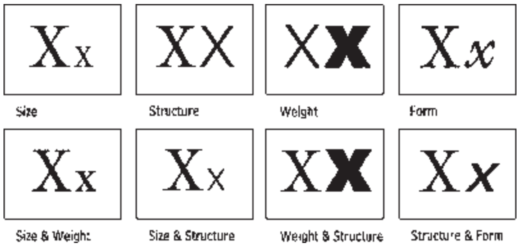


FIGURE 14.18 Typographic contrasts. This figure, which builds on the relationships of contrasts discussed and illustrated in *Design with Type* by Dair (1967), shows typographic contrasts that can be easily used to build information hierarchies.

14.13.10 Black on White versus White on Black and Dark on Light Background versus Light on Dark Background

14.13.10.1 Positive and Negative Type

White on black (or light on a dark background) is generally regarded as less legible and much more difficult to read over large areas. To the eye, white letters on a black background appear smaller than their reversed equivalent. The amount of contrast between the color of type and the background is an especially important factor for online communication. Color adds exponential levels of complexity to these considerations since displays are inconsistent from one situation to another (Figures 14.17 and 14.18) (Dair 1967).

14.14 DESIGN PRINCIPLES: PAGE DESIGN

Typography deals with legibility and *page design* focuses on readability, the ability to read and comprehend information. Can the user find what is needed on the page? The two important functions of page design are motivation and accessibility. A well-designed page is inviting, drawing the eye into the information. Users are motivated to accept the invitation. An effective page design ensures that the reader continues by increasing the ease of understanding and accessibility of the information. (For purposes of simplicity, the term *page design* is used interchangeably to mean page, screen, and document design.) Motivation and accessibility are accomplished by providing the reader with ways to

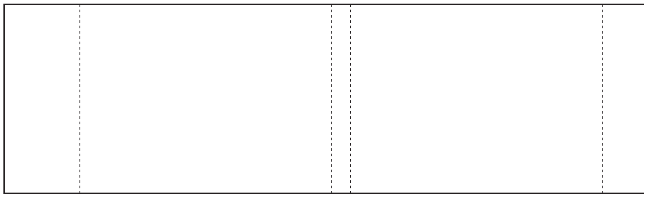


FIGURE 14.19 Grid. The lines that appear in this rectangle provide an understanding of what the grid, or underlying structure, is of this page. The grid is used as a guide to create more pages that use similar relationships for placement and alignments. It is a point of departure for one who understands the system, to create variations, be a bit more playful, yet still provide a consistent “feel” to the user.

quickly understand the information hierarchy. At a glance, the page design should reveal easy navigation and clear, intuitive paths to discovering additional details and information. This is called “visual mapping.”

A page, site, or product visually mapped for easy navigation has

- An underlying visual structure or grid, organizational landmarks, graphic cues and other reader aids (Figure 14.19)
- Distinctly differentiated information types
- Clearly structured examples, procedures, and reference tools
- Well-captioned and annotated diagrams, matrices, charts, and graphics

This kind of visual structuring helps the reader and provides an obvious path through the materials, aids in skimming, gives a conceptual framework, and prevents a feeling of information overload.

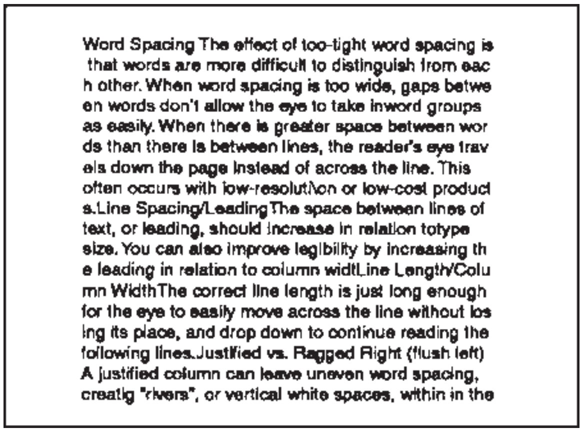
A table of contents is a simple visual map. It quickly provides a general overview of the order and some details about the structure and content. What it does not reveal, however, are priorities. Site maps or other diagrams provide this type of information as well.

14.14.1 Building the Design of a Page

Effective visual mapping is apparent in the sequence shown below that demonstrates the evolution of an accessible page from plain text. As design elements are added, the page becomes inviting to read and content becomes attainable. The final example organizes the content into units of information using line spacing and vertical thresholds or queues. Differentiation in typeface, weight, and scale reinforce structure. Information design techniques, drawn from cognitive science, can be used to improve communication effectiveness and performance.

14.14.1.1 Gray Page or Screen

Raw text interests few readers. When information is presented as a uniform, undifferentiated mass, it is difficult and irritating to use and easy to ignore (Figure 14.20).



Word Spacing The effect of too-tight word spacing is
that words are more difficult to distinguish from eac
h other. When word spacing is too wide, gaps betwe
en words don't allow the eye to take inword groups
as easily. When there is greater space between wor
ds than there is between lines, the reader's eye trav
els down the page instead of across the line. This
often occurs with low-resolut\on or low-cost product
s.Line Spacing/LeadingThe space between lines of
text, or leading, should increase in relation totype
size. You can also improve legibility by increasing th
e leading in relation to column widtLine Length/Colu
mn WidthThe correct line length is just long enough
for the eye to easily move across the line without los
ing its place, and drop down to continue reading the
following lines.Justified vs. Ragged Right (flush left)
A justified column can leave uneven word spacing,
creatig "rivers", or vertical white spaces, within in the

FIGURE 14.20 Gray page or screen.

14.14.1.2 Chunking

Structure the visual field by breaking like kinds of information into manageable groups according to subject matter. Chunks in close proximity are read as related. Graphic devices such as rules and line spaces are used to reinforce a grouping and separate chunks (Figure 14.21).

14.14.1.3 Queuing

Order information chunks visually to reflect the content hierarchy by addressing the user’s requirements of subject matter, order, and importance (Figure 14.22).

14.14.1.4 Filtering

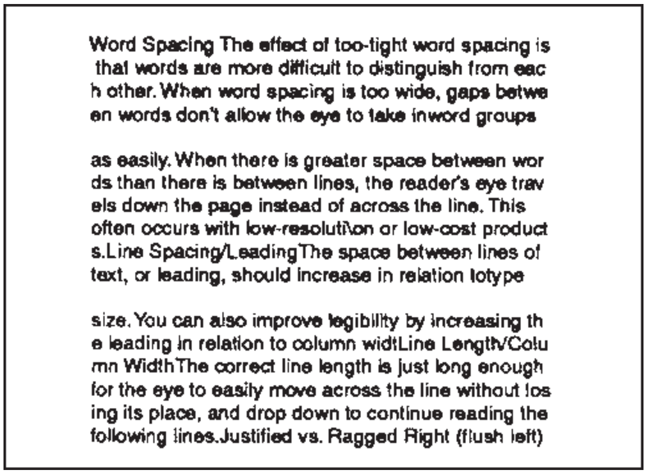
Simplify linguistic and visual order by filtering out unnecessary background noise, which interferes with the information being transmitted. Filtering builds a sense of layers of information by using color, visual cues and symbols, and bulleted lists and headers to make a page effective for a range of users and uses (Figure 14.23).

14.14.1.5 Mixing Modes

People learn through different cognitive modes or styles. Some users favor text, others may prefer illustrations, photos, diagrams, or formulas. To suit these varied learning preferences, information must be translated into several different modes that are then carefully presented to reinforce content and organization (Figure 14.24).

14.14.1.6 Abstracting

The individual page or screen is a microcosm of the complete book, site, or product. The result is a complete codified system of graphic standards, which is effective for both the reader and the producer. Abstracting builds a system of standards that simplifies text organization, creates consistent approaches to preprocessing information, and establishes a customized look for an organization’s products (Figure 14.25).

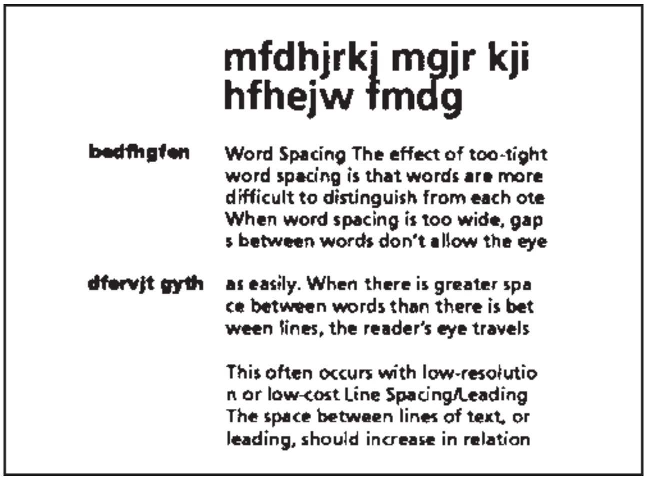


Word Spacing The effect of too-tight word spacing is
that words are more difficult to distinguish from eac
h other. When word spacing is too wide, gaps betwe
en words don't allow the eye to take inword groups

as easily. When there is greater space between wor
ds than there is between lines, the reader's eye trav
els down the page instead of across the line. This
often occurs with low-resoluti\on or low-cost product
s.Line Spacing/LeadingThe space between lines of
text, or leading, should increase in relation totype

size. You can also improve legibility by increasing th
e leading in relation to column widtLine Length/Colu
mn WidthThe correct line length is just long enough
for the eye to easily move across the line without los
ing its place, and drop down to continue reading the
following lines.Justified vs. Ragged Right (flush left)

FIGURE 14.21 Chunking.



mfdhjrkj mgjr kji
hfhejw fmdg

bedfhgfen

Word Spacing The effect of too-tight
word spacing is that words are more
difficult to distinguish from each ote
When word spacing is too wide, gap
s between words don't allow the eye

dfervjt gyth

as easily. When there is greater spa
ce between words than there is bet
ween lines, the reader's eye travels

This often occurs with low-resolutio
n or low-cost Line Spacing/Leading
The space between lines of text, or
leading, should increase in relation

FIGURE 14.22 Queuing.

14.14.2 Other Page Design Techniques

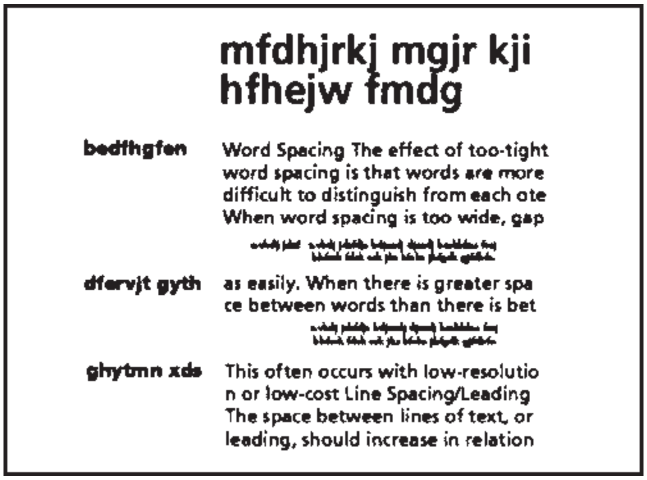
14.14.2.1 White Space

White space (or empty space) is an underutilized but extremely effective design tool. It visually opens up a page, provides focus, helps group like kinds of information, provides resting points for the reader's eye, and creates the perception of simplicity and ease of use.

14.14.2.2 Grid

A grid is a controlled system of organization that allows for the distribution of visual elements in an intelligible order. A grid, as part of an overall design system, provides an underlying structure that determines the horizontal placement of columns and the vertical placement of headlines, text, graphics, and other artwork.

A grid is built on a series of consistent relationships, alignments, and spatial organizations. It acts as a blueprint that can be repeatedly used to create sequential pages, which are related but respond to different content. When the grid



mfdhjrkj mgjr kji
hfhejw fmdg

bedfhgfen

Word Spacing The effect of too-tight
word spacing is that words are more
difficult to distinguish from each ote
When word spacing is too wide, gap

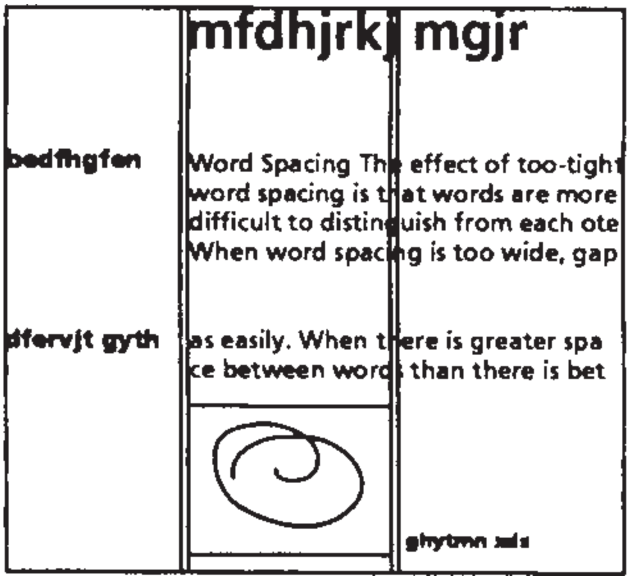
dfervjt gyth

as easily. When there is greater spa
ce between words than there is bet

ghytmn xds

This often occurs with low-resolutio
n or low-cost Line Spacing/Leading
The space between lines of text, or
leading, should increase in relation

FIGURE 14.23 Filtering.



mfdhjrk mgjr

bedfhgfen

Word Spacing Th effect of too-tigh
word spacing is t at words are more
difficult to distin uish from each ote
When word spac g is too wide, gap

dfervjt gyth

as easily. When t ere is greater spa
ce between wor than there is bet

ghytmn xds

FIGURE 14.24 Mixing modes.

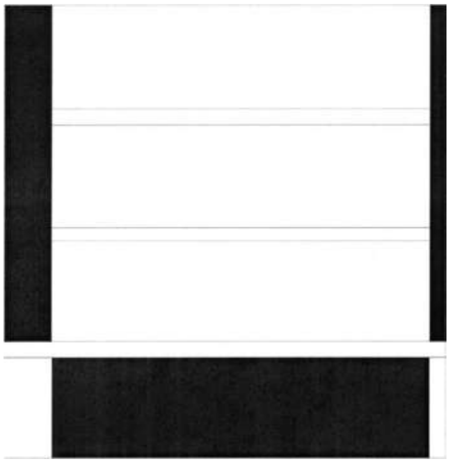


FIGURE 14.25 Abstracting.

system is understood, it forms the basis for consistent application and extension of the design by others who also understand the intention of the system. Every strong design uses an underlying structure or grid to create a consistent look and feel to any form of visual communication. An analogy can be created between the horizontal and vertical lines that compose a grid used on a page and the metal beams that systematically make up the overall supporting structure of a high-rise building. While the structural supports in a building are consistent from floor to floor, the configuration of the space within each individual floor is based on each occupant's needs. The same holds true with the grid on a page. An important tool that improves usability, a grid enables a user to navigate a page quickly and easily. A grid specifies placement for all visual elements. The user anticipates where a button will appear or how help is accessed. Product or program usefulness and ultimately success are greatly increased through the consistency offered by a grid.

14.14.2.3 Field of Vision

Field of vision refers to what a user can see on a page with little or no eye movement; it is the main area where the eye rests to view most of the page. A good design places key elements in the primary field of vision, reflecting and reinforcing the information hierarchy. Size, contrast, grouping, relationships, and movement are tools that create and reinforce field of vision. The user first sees what is visually strongest, not necessarily what is largest or highest. This is particularly true for online information due to the limitations of page real estate and dense information environments.

These concepts, as well as the strength of peripheral vision, can be experienced when viewing a page that has a banner advertisement or moving graphic.

It is virtually impossible to ignore or focus attention on the primary field of vision when there is winking and blinking elsewhere. In fact, superfluous visual devices reduce information's value by distracting and disturbing the user's desire and ability to focus, read and understand.

14.14.2.4 Proximity

This concept applies to the placement of visual elements physically close to each other so that it is understood that these are related elements.

14.14.2.5 Illusion of Depth

Though the online world exists in two-dimensions, contrast can be used to create the illusion of depth. Contrasts created through size, weight, structure, direction, color, texture, and layering can form cues that reinforce hierarchy by giving the illusion that an element is on top of or in front of another.

14.15 CHARTS, DIAGRAMS, GRAPHICS, AND ICONS

The goal of any visual device is to provide the fastest, most efficient path to understanding ideas and to make these ideas clear and compelling. Useful, effective graphics act like visual shorthand, particularly important when the real estate of the page is limited (Figure 14.26). A good visual eliminates the need for text and communicates across cultures. A bad graphic with an unclear meaning that must be reinforced by a long caption can be worse than none at all. The old clichè, a picture is worth a thousand words, is true only if the picture is efficient and effective. In stressful or difficult situations, people do *not* have time to read or the ability to focus on text and/or complex visuals. Though more difficult to achieve, brevity and simplicity in such cases have greater value. Product users prefer well-designed charts, diagrams, and illustrations that quickly and clearly communicate complex ideas and information. Studies show that visual images are retained long after the reader is finished. Designed correctly, visual images can make information memorable and effective. At a minimum, a powerful illustration or graphic can often improve performance simply because it increases user motivation. Visuals are robust communication tools used to (a) visualize and analyze data; (b) present new or abstract concepts; (c) make physical and technical concepts invisible to the eye; and (d) summarize information efficiently and effectively. Visuals explain and reinforce concepts, relationships and data, making them tangible. Photographs, charts, illustrations, icons, or diagrams become thinking tools. The information is clarified, made easier to evaluate, and has greater impact. Visuals are a very effective way to communicate a message, but choosing the appropriate presentation for a concept is critical to the user's ability to effectively comprehend a message. Understanding the limitations of the display medium is crucial to creating a successful visual.

FIGURE 14.26 Zen calligraphy is an example of the historically close relationship between word and image. The Zen master Hakuin (1768–1865) created this symbol to mean "dead," with additional notes saying "Whenever anyone understands this, then he is out of danger."

14.15.1 Tables, Charts, Diagrams

These three types of graphics are discussed in order of complexity. Tables are the least difficult to create, charts the second most difficult, and diagrams the third. Illustrations, graphics, and other images and visuals are the most complex, require more conceptual and visual sophistication, and may require a consultant to create. When is one more appropriate than the other? Determining which format is the most effective is illustrated in Figure 14.27. In addition to this list, it is important to remember that visual cues such as color, shading, texture, lines, and boxes should be considered redundant cues and only used to provide additional emphasis to support the concept.

14.15.2 Icons and Visual Cues

Icons and other visual cues are a form of visual shorthand, which helps users locate and remember information. Developing an easily understood style that is consistent with the overall program style is not easy. Choose a style that is simple and consistently reinforced throughout a product. More complex and unique symbols and icons can be used if usage takes place over a longer period, allowing product familiarity and learning to take place. The MasterCard logo consists of two intersecting circles. After many years of reinforcement, most people immediately recognize it without accompanying text or other explanation. It is very difficult to create an icon that, without explanation, communicates a concept across cultures. For example, the use of a freestanding rectangular box with an open door flap indicates mailbox or in-box. This kind of mailbox is rarely used today and was never used in Europe where mail is placed in slots or upright boxes. Even the concept of mail delivery can be considered strange. This is a case where meaning had to be learned. Although simple ideas presented as icons are appropriate, a program with many complex concepts using colloquial images can make using a program agonizing for users from other cultures. There is an important difference between an icon and an illustration, though the two concepts are often confused. If an icon must be labeled, it is really an illustration. The icon's value as visual shorthand is lost. Better to use a word or short phrase rather than word and image when screen space is at a minimum.

A successful icon is memorable with minimal reinforcement. If after viewing an icon several times a user cannot remember its meaning, then the icon is valueless and should be eliminated. Icon sets should share a similarity of style (businesslike or playful) and possess formal presentation properties consistent with the overall program or product to which they belong (Figure 14.28).

14.15.3 Illustrations and Photographs

As technology improves, the only limit placed on the use of complex images will be by the designer. The most important consideration is appropriateness of the image for the intended audience. Do not use cartoons for a company brochure, or a low-resolution photograph of a control panel when a line illustration is more effective.

	If you want to show ...	**use a ...**
Groups	Group of related items, with a specific order	Numbered list
Relationships	Relationships and steps involved in a process	Flow chart
	Relationships between categories of ideas	Table
	Relationships of tasks taking place over time	Project plan table
Evaluate/compare	Evaluate items against several criteria	Rating table
	Evaluate items against one criteria	Comparison table
	Compare more than one item to more than one variable	Matrix diagram
	Compare several things in relation to one variable	Bar chart
	Compare the relative parts that make up a whole	Pie chart
Hierarchy	Hierarchical structure of an organization	Organizational chart
Concepts	Concept	Illustration and/or text icons, other graphics
	Abstract concept	Complex images, interactive components

FIGURE 14.27 When to use what graphic.

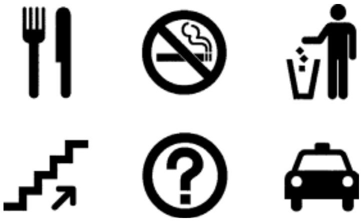


FIGURE 14.28 AIGA/DOT Icons. These six icons, all from the same set, were developed by the American Institute of Graphic Arts and the U.S. Department of Transportation for use in signage. From upper left to lower right: restaurant, no smoking, trash, stairway leading up, information, and taxi.

While understanding the meaning and implications of illustrations and photographs is no easy task, there are guidelines for making choices. A photograph can easily represent an existing object, but issues related to resolution and cross-media publishing can make it unintelligible. If a photograph can be reproduced with proper resolution, cropping and contrast, and emphasize a required detail, then photography is a good choice. A photograph can provide orientation and contextual cues that are more difficult to achieve in an illustration. No matter how simplified or cropped to focus attention, a photograph's reproduction quality is often unpredictable. In this situation, a technical illustration such as a line drawing is more effective. An obvious advantage of illustration is that it can present abstract concepts or objects that do not yet exist, or that may never exist. Another benefit is the ability to focus the viewer's attention on detail. For example, a line drawing can place attention on a specific machine part by changing line weight. To achieve a similar result in a photograph adds time, complicates the image, and possibly never simplifies the explanation.

Regardless of the method used to create a visual explanation, it must clarify and reinforce content. If the purpose of the image is to explain where to locate a piece of equipment, then an overview of the equipment in the environment is appropriate. If the goal is to show an aspect, such as a button location, then the illustration should only focus attention on that aspect. An image can be cropped to focus attention on what is being explained; it depends on the goal of the photograph or illustration.

Situations exist in which it is more effective to use a combination of photography and illustration than either alone. For example, a photograph of an object in its usage environment conveys more information than that of the object itself. If the objective is to show the location of a part of that object, then a line drawing in close proximity to or inset in the photograph is more useful than a photograph or illustration alone.

14.15.4 Guidelines

14.15.4.1 Visuals Should Reinforce the Message

Don't assume that the audience understands how a visual reinforces the argument (Figure 14.29). A clear and concise argument must still be made that helps shorten the process of comprehension and learning and causes the user to say, "Aha, that's how it works together!" Visuals should

- Clarify complex ideas
- Reinforce concepts
- Help the user understand relationships

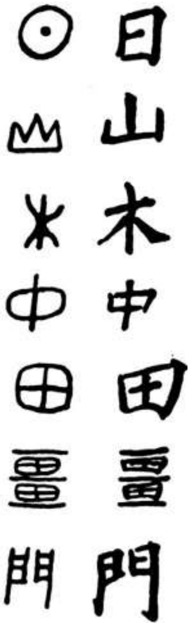


FIGURE 14.29 Thirty centuries of development separate the Chinese ancient characters on the left from the modern writing on the right. The meaning of the characters is (from top to bottom): sun, mountain, tree, middle, field, frontier, door.

14.15.4.2 Create a Consistent Visual Language

Create a consistent visual language that works within the entire communication system. Graphics attract attention. When the user sees a screen, the eye automatically jumps to a visual, regardless of the fact that it may interrupt reading. Graphics should conform to all elements on a page. Unharmonious graphics impede comprehension by increasing the effort needed to understand the relationship between the text and the visual.

14.15.4.3 Consider Both Function and Style

It is important to consider function versus decoration. Albeit wonderful to see an artistically illustrated tax form, is it appropriate to the content or image of both the message and organization it represents? The best graphic is appropriate to the context of the communication and reinforces and validates the message (Figure 14.30).

14.15.4.4 Focus on Quality versus Quantity

Graphics are only effective if they are carefully planned, executed, and used sparingly. A well-considered diagram with a concise caption is more effective than several poorly thought-out diagrams that require long explanations.

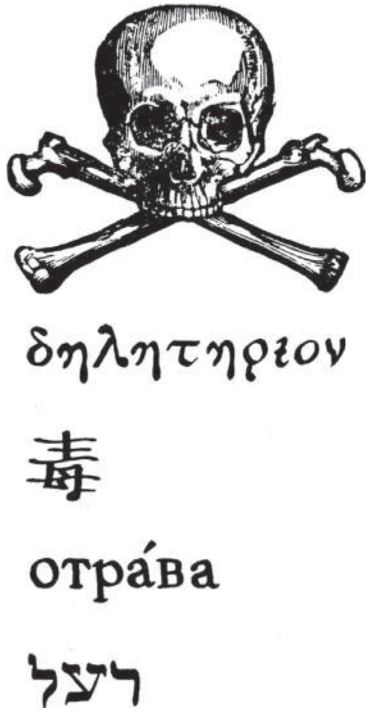


FIGURE 14.30 It is obvious which of the above examples communicates an important message most quickly. The goal for the designer is to communicate the message in the most direct way, so that the user can understand and make decisions based on that information. Obviously some situations are more critical than others, but it is no less important to begin design with consideration of the needs of users.

14.15.4.5 Work with a Professional

Many individuals within an organization can write an internal report but a writer or public relations firm is usually commissioned to find the most effective, relevant, and interesting way to communicate a public message such as presented in an annual report or company brochure. Similarly, a designer or visual communications firm should be retained to oversee the development of user interfaces, graphics, and other visual elements that impact the look and feel and ultimately the overall success of a program.

14.15.4.6 Build a Library to Create Visual Consistency, Organizational Identity, and a Streamlined Process

Because graphics require a professional, they can be very time consuming and expensive to create. Once a visual language and style are established, start building a graphics library. If concepts are repeatedly illustrated, streamline the development process by collecting the supporting illustrations and making them available for reuse. An organizational style can be created for these visual explanations. Through repetition, users can learn to associate a style and method of explanation with an organization, which aids in understanding and reinforces product brand and identity.

14.15.4.7 Reinforce Shared Meaning (Common Visual Language)

A serious issue to consider when creating graphics, particularly conceptual diagrams, is shared meaning, whether it be across an organization or the globe. Individuals can interpret the same diagram in a variety of ways based on backgrounds and experiences.

Truly effective graphics require extra time and effort, but the payoff is tremendous. Graphics are invaluable tools for promoting additional learning and action because they reinforce the message, increase information retention, and shorten comprehension time.

14.16 COLOR

Though color should be considered a reinforcing, or redundant, visual cue, it is by far the most strongly emotional element in visual communication. Color evokes immediate and forceful responses, both emotional and informational. Because color is a shared human experience, it is symbolic. And like fashion, the perception of color changes over time. In all communication, color can be used to trigger certain reactions or define a style. For example, in Western business culture, dark colors such as navy are generally considered conservative, while paler colors such as pink are regarded as feminine. In other cultures, these color choices have an entirely different meaning.

The appropriate use of color can make it easier for users to absorb large amounts of information and differentiate information types and hierarchies. Research on the effects of color in advertising show that ads using one spot of color are noticed 200% more often than black-and-white ads, while full-color ads produce a 500% increase in interest. Color is often used to

- Show qualitative differences
- Act as a guide through information
- Attract attention/highlight key data
- Indicate quantitative changes
- Depict physical objects accurately

All in all, color is an immensely powerful tool. Like the tools of typography and page design, it can easily be misused. Research shows that while one color, well used, can increase communication effectiveness, speed, accuracy, and retention, multiple colors when poorly used actually decrease effectiveness. Because it is readily available, it is very tempting to apply color in superficial ways. For color to be effective, it should be used as an integral part of the design program, to reinforce meaning and not simply as decoration. The choice of color—while ultimately based on individual choice—should follow and reinforce content as well as function.

14.16.1 Basic Principles of Color

14.16.1.1 Additive Primaries

The entire spectrum of light is made up of red, green, and blue light, each representing a third of the spectrum. These three colors are known as "additive primaries," and all colors are made up of varying amounts of them. When all three are combined, they produce white light.

14.16.1.2 Subtractive Primaries

If you add and subtract the three primaries, cyan, yellow, and magenta are produced. These are called "subtractive primaries."

Green + Blue – Red = Cyan
Red + Blue – Green = Magenta
Red + Green – Blue = Yellow

Color on a computer display is created by using different combinations of red, green, and blue light. In print, colors are created with pigments rather than light. All pigments are made up of varying amounts of the subtractive primaries. The three attributes of color are

1. ***Hue***—the actual color
2. ***Saturation***—the intensity of the color
3. ***Value***—includes lightness and brightness

"Lightness" refers to how light or dark a color appears. "Brightness" is often used interchangeably with lightness; however, lightness depends on the color of the object itself, and brightness depends on the amount of light illuminating the object.

14.16.2 How to Use Color

14.16.2.1 Less Is More . . . Useful and Understandable

Just as you can overload a page or screen with too many typefaces, you can have too many colors. Given the unpredictability of color displays, users, and viewing situations, the choice can get very complicated. Color is often best used to highlight key information. As a general rule, use no more than three colors for primary information. An example is the use of black, red, and gray—black and red for contrasting information, gray for secondary. When thinking about color online, one must remember that each display will output color in a different way. Add to that the lighting situation and a variety of users. All these factors affect color choice.

14.16.2.2 Create a Color Logic; Use Color Coding

Use a color scheme that reinforces the hierarchy of information. Don't miscue the audience by using different colors for the same elements. Whenever possible, try to use colors that work with the project identity or established visual language. Create a color code that is easily understood by the user and reinforces the information.

14.16.2.3 Create a Palette of Compatible Colors

Harmonious color is created by using a monochromatic color scheme or by using differing intensities of the same hue. However, make them different enough to be easily recognized and simple enough to be easily reproduced, no matter what medium you are using.

14.16.2.4 Use Complementary Colors with Extreme Caution

These are colors that lie opposite each other on the color wheel. Let one dominate and use the other for accents. Never place them next to each other because the edges where they meet will vibrate. Though this was the goal of pop art in the 1960s, it makes pages impossible to read. One must check each particular display, as the calibration of monitors can unexpectedly cause this to happen.

14.16.2.5 Decisions Regarding Color in Typography Are Critical

Colored type appears smaller to the human eye than the same type in black. This is important to consider when designing user interfaces. One must also consider the "smear" effect on typography in displays, based on the color chosen and interaction with colors around it. Additionally, quality and calibration of displays impact characteristics of color online.

14.16.2.6 Consider the Viewing Medium

The same color looks different when viewed on different viewing media such as a computer display, an LCD projector, color laser printer versus dot-matrix output, glossy versus dull paper.

14.16.2.7 Context Is Everything

Though printed color is very familiar and more controllable, projected color is inconsistent and varies depending on such things as lighting, size of the color area, size and quantity of colored elements, lighting, and output device. One must check all output/viewing possibilities to insure that a color is readable as well as legible, and not depend on cross-media specification for insuring consistency. What might look good on a laptop may not be readable when projected in a room for hundreds of people to view, and may look completely different when printed in a corporate brochure. The amount of color will affect how it is viewed as well as the best background choice. A blue headline is very readable on a white background, but if that background becomes a color, then readability can be reduced dramatically, depending on how it gets presented on each particular display.

14.16.2.8 Contrast Is Critical When Making Color Choices

Contrast is the range of tones between the darkest and the lightest elements, whether one is considering black and white or color. The desired contrast between what is being "read" (this includes graphics, photographs, etc.) must be clearly and easily differentiated from the background it is presented against. If there is not enough contrast (of color, size, resolution, etc.), it will be difficult or impossible to read. This is particularly a problem with online displays, as the designer has no control of quality of the output display.

14.16.2.9 Quantity Affects Perception

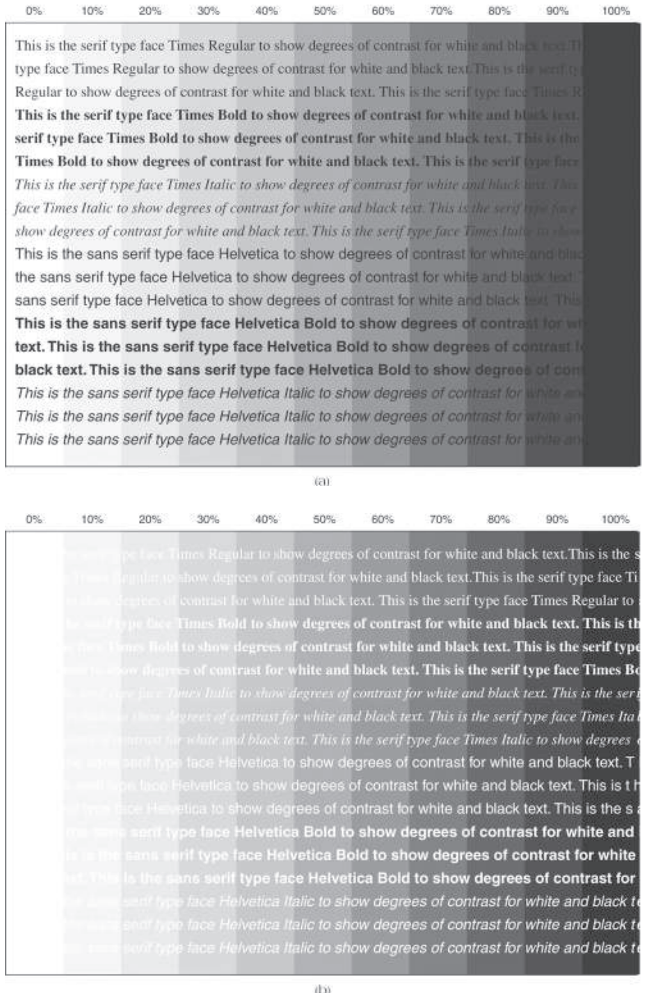
A small amount of color will be perceived differently than the same color used in a large quantity. In the smaller area, the color will appear darker; in the larger area, the color will appear lighter and brighter.

14.16.2.10 Use Color as a Redundant Cue When Possible

At least 9% of the population, mostly male, is color-deficient to some degree, so it is generally not a good idea to call out warning points only through color. With a combination of color and a different typeface, and so on, you won't leave anyone in the dark.

14.16.2.11 We Live in a Global World, So When in Rome . . .

Remember that different colors have different connotations within various cultures, religions, professions, and so on. For example, in the United States on February 14, red means love, but in Korea, red means death, and in China, red is used in weddings and symbolizes good luck and fortune. In many other countries, red means revolution. To a competitor, red means first place, and to an accountant, red means a negative balance. To a motorist, red means stop, and in emergencies, a red cross means medical help (Figure 14.31).



(a)

(b)

FIGURE 14.31 Trying Examples in Your Context. Since color is not available in this particular edition, try your own experiment. Take a look at this illustration, and recreate a paragraph of text, with the background graded from 100% to 0%, choosing one color for the background. Then set lines of type in a variety of typefaces and sizes, to see where it becomes legible or totally impossible to read. The important thing to remember is to test out whatever choices you make within the particular context and parameters, including viewing/projecting devices. Such things as lighting, projection distance, and users' physiological constraints can make all the difference as to whether something can be read or not.

14.17 CREATING A SYSTEM: GRAPHIC STANDARDS

With the explosion of new publishing media in a global marketplace, the need for guidelines for developing and producing consistent, quality communication has taken on a new urgency. New technologies make it easy to generate images, offering a wealth of options for experienced and inexperienced publishers alike. The danger lies in creating visual chaos, with every element demanding attention beyond the point of sensory overload. With new tools, chaos can happen faster, at a lower cost, and with greater distribution. A graphic standards manual prevents this confusion.

A *graphic standards manual* is the physical manifestation of an identity system. The design historian and educator Meggs (1998) writes in *A History of Graphic Design* that identity systems arose in the 1950s with the rise of multinational corporations that began to recognize the value and power found in presenting a cohesive visual image globally. A quality-control agent, a standards manual allows an organization to document guidelines and provide tools for organizing and structuring communications and reinforcing brand identity to diverse internal and external audiences. It explains the methodology behind the design, specifies written and visual language, production materials, and methods, and gives examples of how to and not to use the identity system so that standards can be implemented by different people, in different places, at different times. A standards manual supports expansion by explaining how to maintain a consistent brand and organizational look and feel as new products, features, and technology are introduced.

A graphic standards system provides

- Built-in quality: The system ensures that the correct organization/product image is communicated to all audiences. Standards promote consistency in handling information across product lines, divisions, projects, and so on.
- Control over resources: A system provides dramatic managerial control over resources that use time, money, and materials. Well-developed standards build in flexibility. New communications are

easily developed without the original designer and in many instances are developed within the organization itself.

- Streamlined development process: A graphic standards system helps structure thinking for content, design, and production by providing a guideline of predetermined solutions for communication problems. Typical problems are solved in advance or the first time they occur. Most importantly, a graphics standards system encourages the organization as a whole to progress to higher-level issues of communication effectiveness.

14.17.1 What Does a System Cover?

Graphic standards historically have been applied to an organization's logo, stationery, business cards, and other printed materials. As the online portion of an organization's identity dominates, providing for cross-media guidelines is even more critical. Graphic standards are generally communicated to the organization in print and electronic form. Documentation often includes

- Corporate identity manuals: Style guides available in print and online illustrate the application of the standards across the company's publications and provide specifications for production and expansion.
- Templates and guidelines: Templates and guidelines are available in paper and electronic form and are used to develop pages for both environments.
- Editorial style guides: Editorial guides determine the use of product/service names, punctuation, spelling, and writing styles.

14.17.2 Developing the System

When developing a corporate graphics standards system, consider the global publishing needs of the company, the resources available for producing documents, and the skill level of those in charge of production. To responsibly determine overall needs, a team effort is required. Personnel from areas such as information systems, graphic design, usability, and marketing along with engineers, writers, and users should be involved in the process. This team approach helps build support for, and commitment to, the corporate standards. The development of a comprehensive system follows the information design process of audit, development, and implementation.

14.17.3 Audit

The audit is a critical step in determining the scope and parameters of an organization's corporate graphic standards. Specific questions for the audit phase include

- What is the purpose?
- Who are the audiences?
- What are the differences and similarities between audiences?
- Who will do the work?
- How long will it take?
- What tools will be used?
- What is the desired company or product image?

14.17.4 Development

Goals for the development phase include: (1) the design of standards that are easy to read, use, and project a consistent corporate image; and (2) design of products that fit within the production parameters of the company.

14.17.5 Implementation

The implementation phase must ensure that the system is accepted and used properly. This requires training and support, easy procedures for distributing and updating materials, and a manual explaining how to use the system. The development of standards is in itself an educational process. It requires all participants to be aware of communication objectives and what is needed to meet them. As alternatives are developed and tested, management has the opportunity to evaluate its company's purpose, nature, and direction as well as its working methods and communication procedures. The process requires commitment and involvement across many departments and levels. The result is an empowering of the organization—planting the seeds for growth and increased effectiveness.

14.18 DESIGNING THE EXPERIENCE

The heart of interface design is to define and create the user's experience; what it is really like for people facing the monitor, using a cell phone, or an ATM. Though presentation possibilities are expanding day by day, our capacity to understand, use, and integrate new information and technology has not grown at the same rate. Making the most appropriate media choices, whether image, animation, or sound, to explain complex ideas to widely varied audiences is no easy task. The most important guideline is to understand that there are no rules, only guidelines. It is a generalization to say that a visual principle works a certain way because any change in context changes the application of the principle. For example, in the early days of the software industry, research showed that a specific blue worked well as a background color. Now, however, depending on monitor calibration, as well as environmental lighting, that blue could be a disaster. In fact, that blue can often vibrate if type in particular colors is placed on it. Of course, it depends on the type quantity, size, weight, viewing situation, and so on. Sound complex? For this reason, it's important to understand the principles, test the ideas, and then test results on every output device that will be used. Putting known guidelines together with experience continually gathered from the field allows the designer to develop a clear understanding of what works well in a well-defined environment and user situation. The next key guideline is to keep it simple. Although many tools are available, there is only one goal: to clearly communicate ideas. The designer must always ask, "What is the most efficient and effective way to communicate this idea?" A good illustration might work better (and take less bandwidth) than an animated sequence. Text set in a simple bold headline might allow the user to read the page more efficiently than text placed in a banner moving across the top of a page that constantly draws the eye upward. Animated icons are entertaining, but are they appropriate or necessary for serious financial information? It's tempting to use new tools. The best tip is to use a tool only if it can explain an idea better than any other tool, enhance an explanation, or illustrate a point that otherwise could not be made as effectively or efficiently. The best design is not noticed; it just works. Products are used to accomplish tasks, *not to draw attention to the design*. The best test of product success is the ease with which a user can understand and complete a task and move on to the next task. Real estate and online real estate are alike in that they both stress location, location, location! With such a premium on space, and so much to accomplish in so little time, be considerate and efficient with online real estate.

Use the elements found in a product's graphic standards appropriately. Constantly consider choices and context and review design principles. The following are issues and considerations to continually evaluate when presenting interactive information.

14.18.1 Effective and Appropriate Use of the Medium

Transitioning a print document to an online environment requires rethinking how the document is presented. Viewing and navigating through online information requires radically different design considerations and methods. Users do not necessarily view the information in a linear way, in a specific order, or timeframe. Interactive media viewed on computer screens have quite different characteristics and potential, particularly as information crosses platforms, resolutions, and environments. The rich medium of print allows a book—a product—to be held, viewed, and read in a sequence determined by the user. The physicality of a book provides sensory cues that are not present on a two-dimensional monitor. Interface designers must find ways to provide equivalent cues that encourage people to handle products comfortably and with confidence.

14.18.2 Element of Time

The element of time is the critical difference between static and interactive media. The sense of interaction with a product impacts the user's perception of usefulness and quality. Animated cues such as blinking cursors and other implied structural elements like handles around selected areas become powerful navigational tools if intuitively understood and predictably applied. Consider how the product will be used. Will the user sit down and calmly use the product or will he or she panic and fumble with a keypad? Will the task be completed at one time or at intervals over hours, days, months, and years? The element of time contributes to the design criteria and choices.

14.19 CONSISTENT AND APPROPRIATE VISUAL LANGUAGE

A major issue is the unpredictability and vastness of products. Providing way-finding devices that are easy to recognize, understand, and remember, include

- Clear and obvious metaphors
- Interface elements consistent with the visual style of other program parts, including consistent style for illustrations, icons, graphic elements, and dingbats
- Guidelines for navigational aids such as color, typography, and page/screen structure that are consistent with other parts of product support

14.19.1 Navigational Aids

Progress through a book is seen and marked in many ways. Bookmarks and turned corners serve as placeholders. Pens act as mnemonic devices highlighting or underlining text. Table of contents and indexes reveal content location. A finger marks a passage to be shared with a colleague as a book is cradled.

Unlike a book, a digital document or program cannot be seen or touched in its entirety. If a document cannot be held, how is specific location known in relation to overall location? How do users return to or move forward through content? How do users travel through unfamiliar space?

Navigational aids provide users with highways, maps, road signs, and landmarks as they move through the online landscape. They enhance discovering and communicate the underlying structure; thereby providing a sense of place so that users know where they are, where they have been, and how to move elsewhere or return to the beginning. Using or building on already familiar visual elements, such as those found in other products and earlier releases, leverages existing knowledge. Graphic standards support this as well. Consistent use of page layout and grid structure makes it easier to remember how information is organized and where it is placed or zoned. This ensures that whatever visual cues are applied can take advantage of the user's experience, ultimately saving time for both the designer and user.

14.19.2 Graphics/Icons

Visual representations such as site maps, graphics, and icons are effective devices for orienting users within a program. A site map offers an overall product view and shows sections or units of information and how these units are related. In the digital environment, graphics and icons assume the role of contents, indexes, and page numbers and can be more effective guides through and around a program than their counterparts in print, because tools such as roll-overs highlight functionality. Creating effective graphics and icons requires that intent and action are defined and designed.

14.19.3 Metaphor

Prior knowledge makes it easier to learn because it provides a conceptual framework on which information can be associated and expanded. When it was first introduced, the metaphor of a desktop with a filing system for a software interface that organized data in a program was easy to grasp because it built on a known experience. Using familiar visual analogies helps users easily understand and organize new information.

14.19.4 Color

Color is a free and very seductive design tool once a monitor is purchased. Use it intelligently. A monitor offers limited workspace. Color can replace or reinforce written explanations when meaning is assigned to it and it is applied methodically. A blue background is always utilized in a testing section and a yellow background in a section overview.

14.19.5 Legibility

Legibility is the ability to read information on the page. The page can be a screen, and as such, has special considerations. Color, size, background, movement, viewing environment, lighting, and resolution play a critical part in legibility.

14.19.6 Readability

Readable screens demand intelligent visual representations and concise, unambiguous text. Meaning can be implied or inferred by the placement of elements in designated areas or zones reserved for distinct information. This makes optimal use of a limited space and increases comprehension and accessibility.

14.19.7 Guidelines

14.19.7.1 Use the Analogy of a Poster as a Guide to Design

A home page is the equivalent of an attention-grabbing poster unpredictably placed in uncontrollable locations. Unlike a home page were a mouse-click on a speaker's name can give biographical information and a click on location can find directions, a poster's static format restricts the amount and depth of information that it can offer.

In print, information is presented in a fixed order. On the web, information is organized hierarchically in a manner that is radically different. Individual users can access the information offered on a website in a sequence that suits their intents and purposes. Online environments offer little regulation over how and in what order the product is accessed. While designers can make suggestions and guesses, this lack of control requires fundamental differences in information presentation. A well-designed home page, like a well-designed poster, should hint at all topics contained in the site, provide high-level information about these topics, and suggest easy paths to access this information. If information goes beyond a single screen, its design must visually communicate location through strong visual hints, so that the user investigates beyond what is immediately visible. Imagine the design considerations required for smaller, handheld, voice-activated devices.

14.19.7.2 Design for the Most Difficult Common Denominator

Design the interface in anticipation of a worst-case scenario. If a manager will use a product in a quiet office with a fast connection, perfect lighting, and a large monitor, then the problem is different from that of a contractor accessing critical information on a laptop in the field. User profile is often unknown because *new technologies define new categories as new opportunities are recognized*. Consider the breadth of possibilities. Design from the user's perspective. Testing, viewing, and questioning can make the difference of product acceptance or not.

14.19.7.3 Avoid Overuse of Saturated Colors

Saturated colors such as red tend to jump out at the viewer and can be distracting and irritating. Red is usually not a good choice for large areas of on-screen color. High impact is dependent on the contrast between background and foreground colors. On a black background, both yellow and white have a higher impact than red. Consider variations in every viewing situation including how these variations affect contrast among page elements and overall legibility and readability.

14.19.7.4 Consider Different Users' Levels of Skill

Navigational tools should be simple enough for a novice to use, but should not impede an expert. Detailed visual maps and other graphics should be available for those who need them, without hindering an experienced user who wants to bypass an explanation.

14.19.7.5 Be Aware of the Fatigue Factor

Although there is no definitive answer on fatigue caused by looking at a computer screen for long periods of time, it is a central factor to consider. According to *Color and the Computer*, by H. John Durrett, looking at a well-designed computer screen should not cause any more fatigue than reading a book or writing a report. Though some would disagree with this statement, many people spend more time with their computer than a book and no doubt could offer additional opinions on this subject. As interactive media becomes a commodity, the focus will not be on what a program does, but on how it does it. This will make the difference between product acceptance and product failure. Success or failure will be judged by the ease with which a product is used and how easily users perceive its interface.

14.19.7.6 Other Differences to Consider

There are many differences that impact how and why interfaces are designed, and many of these differences are discussed in more detail in other chapters. A designer should never forget that physical and mental impairments impact an ability to read, comprehend, and use interfaces.

14.19.7.7 Use the "Squint Test" to Check the Design

The squint test is a very simple self-test that checks visual hierarchy. Simply squint at the page so that details are out of focus. What is the first, most dominant element on the page? Is this what should be seen first? What has secondary importance? Cognitive psychology calls this "visual queuing." Successful interaction design creates a visual order that the user can easily follow.

14.20 CHALLENGES AND OPPORTUNITIES

14.20.1 Creating Your Own Guidelines

Interactive communication designers face great challenges. How can products that are seen, read, understood, and acted upon be created? Given increasing variety and complexity, how can the power of new technologies be harnessed? How can informed visual choices be made? **WARNING**: No book, seminar, or technology will turn someone into a professional designer! Design is not a craft dependent upon aesthetic ability. Design requires education, training, and experience in a variety of related disciplines equivalent to that of an architect, engineer, surgeon, or cabinetmaker. The following guidelines are offered as starting points, first steps in understanding how to make informed design decisions—design that provides the best, most thoughtful, and appropriate integration of both form and function.

- There are no universal rules, only guidelines. If there were rules, everything would look the same and work perfectly according to these rules. Each situation is different with its own context and parameters.
- Remember the audience: be a user advocate. Think about audience needs first throughout the development process. Who is in the audience? What are their requirements? How and where will the audience use the product? The evaluation criteria used in the design-development process springs from the answers to these and other questions. Designers must understand and advocate for the user.
- Structure the messages. Analyze content to create a clear visual hierarchy of major and minor elements that reflects the information hierarchy. This visual layering of information helps the user focus on context and priorities.
- Test the reading sequence. Apply the squint test. How does the eye travel across the page, screen, or publishing medium? What is seen first, second, and third? Does this sequence support the objectives and priorities as defined in the audit?
- Form follows function. Be clear about the user and use environment first. An effective interface design represents and reinforces these goals.
- Keep things simple. Remember the objective is to communicate a message efficiently and effectively, so that users can perform a task. Fewer words, type styles, and graphic elements mean less visual noise and greater comprehension. An obvious metaphor enhances intuitive understanding and use. The goal is to transfer information, not show off features or graphics.
- People don't have time to read. Write clearly and concisely. Design information in an economical, accessible, intuitive format that is enhanced by a combination of graphics and typography. Graphics, if well thought out and designed as an integral part of the page, are very powerful and can efficiently and effectively provide explanations while saving space on a page.
- Be consistent. Consistent use of type, page structure, and graphic and navigational elements creates a visual language that decreases the amount of effort it takes to read and understand a communication piece. The goal is to create a user experience that seems effortless and enjoyable throughout.
- Start the design process early. Don't wait. Assemble the development team of designers, usability professionals, engineers, researchers, writers, and user advocates at the beginning of the process. With interactive media, the traditional review and production process will change. The process is less of a handoff and more of a team effort; it's more like making a film than writing a book. Successfully applying the principles of good design enables an organization to communicate more effectively with its audiences and customers, improving the worth of its products and services and adding value to its brand and identity.
- Good design is not about good luck. Good design for usable interfaces appropriately applies the fundamentals of visual design to interactive products. Creating the most useful, successful design for an interactive product is difficult. The design process is iterative and experiential. There are usually several possible ways to solve a problem, and the final design decision is dictated by the best choices that work within the parameters at any particular time. Advocate on behalf of the user. Users are why designers are here and have this work to do. Users are everywhere, often in places not yet imagined. As the world grows smaller and becomes even more connected, the opportunity lies in where and what has not been discovered.

REFERENCES

Berry, J. D., ed. 2004. *Now Read This: The Microsoft Clear Type Font Collection*, 15–17, 61–69. Microsoft Corporation.

Bringhurst, R. 2005. *The Elements of Typographics Style*, 17–24. Point Roberts, WA: Hartley & Marks Publishers.

Craig, J. et al. 2006. *Designing with Type: The Essential Guide to Typography*, 23–26. New York: Watson-Guptill Publication.

Dair, C. 1967. *Design with Type*, 49–70. Toronto, ON: University of Toronto Press.

Davis, M., P. Hawley, B. McMullan, and G. Spilka. 1997. *Design as a Catalyst for Learning*, 3–12. Alexandria, VA: Association for Supervision and Curriculum Development.

Donis, D. A. 1973a. *A Primer of Visual Literacy*, ix. Cambridge, MA: MIT Press.

Donis, D. A. 1973b. *A Primer of Visual Literacy*, 22–23. Cambridge, MA: MIT Press.

Felici, J. 2003. *The Complete Manual of Typography: A Guide to Setting Perfect Type*, 29. Berkeley, CA: Peachpit Press.

Meggs, P. B. 1998. *A History of Graphic Design*, 363. New York: John Wiley & Son.

Re, M., ed. 2002a. Reading Matthew Carter's letters. In *Typographical Speaking: The Art of Matthew Carter*, 20. Baltimore, MD: Albin O. Kuhn Library & Gallery.

Re, M., ed. 2002b. Reading Matthew Carter's letters. In *Typographical Speaking: The Art of Matthew Carter*, 23. Baltimore, MD: Albin O. Kuhn Library & Gallery.

Re, M., ed. 2002c. Reading Matthew Carter's letters. "Type Vocabulary." In *Typographical Speaking: The Art of Matthew Carter*, 82. Baltimore, MD: Albin O. Kuhn Library & Gallery.

Spencer, H. 1969. *The Visible Word*. New York: Hastings House.

Tracy, W. 1986. *Letters of Credit: A View of Type Design*, 30–1. Boston, MA: David R. Godine Publisher.

15 Globalization, Localization, and Cross-Cultural User-Interface Design

Aaron Marcus and Emilie W. Gould

CONTENTS

15.1 OVERVIEW

User-interface (UI) design enables people around the world to access data and functions. Issues of effective communication for diverse sets of users can take various forms: internationalization of inputs and outputs, translation of text and messages, revision of graphics and text to localize the product/service, and implementation of new communication approaches, including cognitive and rhetorical techniques. Often, appropriate localization of the UIs of products/services combines a mixture of partially universal and partially local solutions, depending on a needs analysis of the users and a business justification for globalization.

As in all good design, global UIs require attention to the product/service UIs essential metaphors, mental models, navigation, interaction, and appearance. In addition, the UI development process must also pay special attention to the user group and its culture. Cultural dimensions, that is, those patterns of behavior and thinking proposed by anthropologists and other analysts of culture, can provide insight and help designers adjust UIs to better serve users. By recognizing similarities/differences and managing users' experiences of familiar structures and processes, the UI designer can achieve more compelling and successful solutions. Facilitating users' engagement with products and services to enhance their productivity and enjoyment should be the primary design objective. Paying attention to culture can assist.

Culture should be considered from the start of the design process. If the functions and data are likely to be of value to target populations outside of the original market, it is usually worthwhile to plan for international and intercultural factors during initial development, so that the product/service can later be efficiently customized (e.g., placing text in separate files for easy translation). Rarely can a product achieve worldwide acceptance with a "one-size-fits-all" solution. Making products ready for global use generally increases international sales. However, in some countries, monolithic domestic markets inhibit awareness of, and incentives for, globalization. For example, because the United States has been such a large producer and consumer of software in the past, it is not surprising that some U.S. manufacturers still focus only on domestic users. However, as U.S. industries like moviemaking and games development have discovered, non-U.S. sales can be a significant portion of total sales or may be even larger than domestic sales. In addition, middle-class consumer markets have arisen in China and India that are larger than the entire populations of the United States or European Union. Consequently, more and more software developers of products/services are considering international and domestic markets as part of the initial design. This change of attitude leads to greater interest in and involvement with cross-cultural communication issues.

Of course, globalization tends to increase initial development costs. Consequently, some software products are initiated with international versions (e.g., typically 5 to 7 languages for global products originating in the United States, or 10 to 15 languages for European Union websites) that are released in sequence because of limited development resources. As needs or opportunities arise, other products are "retrofitted" to suit the needs of a particular country, language, or culture. However, ad hoc solutions often fail because of the lack of original planning for globalization. Good planning minimizes globalization costs by building in the infrastructure for internationalization, translation, and localization.

At first, global enterprises distributed computer-based products and services with minimal changes to their most important international markets to achieve cost-efficient production, maintenance, distribution, and user support. However, the concept of modifying software and hardware to accommodate national conventions for input and output has been considered since at least the early 1980s when such companies as IBM (at its IBM Toronto lab) began identifying software requirements for internationalization. In the early 1990s, a number of software-related working groups focused on overcoming the obstacles to a multilingual World Wide Web (Nicol 1994). A few years later, the growing disparity in access to computers (and the Internet) between people in developed and developing countries sparked discussion of a "digital divide." Stephanidis, among others, promoted the availability of appropriate UIs for all peoples in all countries worldwide regardless of ability, disability, or culture and produced significant compilation documents advocating a "universal access" or "user interfaces for all" approach (2000, 2009). By the beginning of the twenty-first century, Warschauer (2003) noted that the digital divide was no longer just about access to technology; rather, technology must be successfully integrated into the lives of people from developing countries and allow them to engage in socially meaningful practices. It has become increasingly important, technically viable, and economically necessary to produce localized versions of programs, web pages, and applications for specific global markets. Consequently we are now at a point when UIs should be designed for specific user groups, not merely translated and given a superficial "local" appearance for quick export to different countries.

Insufficient attention to global UI design can lead to undesired, sometimes embarrassing, and sometimes critically misunderstood communication. Inexpert internationalization of data fields makes it difficult or impossible for international customers to enter shipping addresses. Anecdotes of poor translation abound in books about international marketing. For example, it has been claimed (though not fully proven) that Pepsi's slogan "Come alive with Pepsi" was interpreted in China as, "Pepsi brings your ancestors back from the grave" (Hendrix 2001; Snopes.com 2007). However, there are several well-documented situations where differences of culture have had significant business implications. In 2001, Saudi Arabia's Higher Committee for Scientific Research and Islamic Law banned Pokémon video games because they felt the cards "possessed the minds" of Saudi children, included Christian and Zionist symbols, and "resembled a game of gambling," thus closing off one of the Middle East's largest markets to Japanese Nintendo's multibillion dollar enterprise (Associated Press 2001; BBC, 2001). More recently, a Saudi court convicted a Lebanese man of sorcery for telling horoscopes (Death Penalty News 2010), which prompted one software company to remove an icon representing a "magic wand" for fear of offense.

By contrast, effective internationalization of input and output data fields facilitates international sales. Appropriate language translation leads to greater comprehension, and customer service costs drop when instructions are displayed in users' native languages. Moreover, allowing users/customers to pick their language leads to greater attention and retention. This implication is especially significant for web-based communication and commerce, where Forrester Research (1998) first reported that visitors remain twice as long reviewing local-language sites as they do English-only sites, and business users are three times more likely to buy when communication is in their own language. Forrester Research Inc. (2009) continues to make the case that internationalization, translation, and localization are necessary for online retailers to sell internationally.

This chapter discusses the development of global UIs intended for users in many different countries with different languages and cultures. The text presents a survey of important issues, as well as recommended steps in the development of UIs for various platforms (desktop, web, mobile devices, and vehicles), for international and/or multicultural user populations. With the rise of the Internet and application-oriented websites and mobile devices, the challenge of designing good UIs has become an immediate, practical matter, rather than a theoretical issue.

In this chapter, the topic is discussed from a user perspective, rather than a technology or code perspective. The chapter will accomplish the following:

- Introduce fundamental definitions of UI design and globalization processes for design.
- Demonstrate why globalization is vital to the success of computer-based communication products/services.
- Outline critical areas for globalization with specific guidelines.
- Introduce various concepts of culture, particularly cultural dimensions.
- Recommend other issues that relate to culture and design.

15.2 USER-INTERFACE DESIGN AND GLOBALIZATION PROCESSES

15.2.1 Definitions

By managing the user's experience with familiar structures and processes, preferences, expectations, and enjoyment of novel approaches, the UI designer can achieve compelling forms that enable the UI to be more usable, useful, and appealing. The International Organization for Standardization (ISO) in Switzerland defines *usable* interfaces as effective, efficient, and satisfying (ISO 1998). Further effort to globalize UI design, whose content and form are so dependent upon effective input/output, languages, and effective communication, improves the likelihood that computer-based products and services on all platforms (e.g., desktop, mobile, home appliances, and vehicles) will be successful in many different locations globally. Global product distribution requires a strategy and tactics for the design process that infuse international and cultural requirements into product development, marketing, distribution, and maintenance.

From the designer's perspective on a corporate team, the two primary objectives are as follows:

- Provide a consistent UI and, more generally, design a satisfying user experience that extends across all appropriate products and services.
- Design products and services with their necessary support systems that can be appropriately localized, translated, and designed for specific markets.

As in all usability-oriented development, the first objective requires attention to the context, objectives, and goals of the users and to the five essential components of UI designs as defined below. All design efforts must begin with a profound understanding of intended users, who are generally characterized on the basis of their demographics, experience, education, and organizational or leisure roles. Their individual needs and wants, as well as their group roles, define their tasks. User-centered, objective-, and task-oriented design methods facilitate effective UI designs that acknowledge and respect users' goals.

UIs conceptually consist of five components: (1) metaphors, (2) mental models, (3) navigation, (4) interaction, and (5) appearance (Marcus 1995, 1998, 2006).

- *Metaphors*: Essential concepts conveyed through words and images, or through acoustic or tactile (haptic) means. Metaphors embrace both individual items as well as overarching concepts that characterize interaction, such as the specific "trash can" denoting "deletion" within the general "desktop" metaphor, or the "shopping cart" denoting selection of items intended for purchase within a "shopping" website.
- *Mental models*: Organization of data, functions, tasks, roles, and people in groups at work or play. The term is similar to, but distinct from the notion of cognitive models, task models, user models, and so on. A mental model is intended to convey the organization observed in the UI itself, which is presumably learned and understood by users and which reflects the content to be conveyed, as well as the user tasks.
- *Navigation*: Movement through mental models, afforded by windows, menus, dialogue areas, control panels, touch screens, and so on. The term implies dialogue and process, as opposed to structure; it focuses on potential sequences of actions to access dynamic content, not only static content.
- *Interaction*: The means by which users communicate input to the system and the feedback supplied by the system. The term implies all aspects of command-control devices (e.g., finger gestures, keyboards, mice, joysticks, microphones, etc.), as well as sensory feedback (e.g., changes of state of virtual graphical buttons, auditory displays, and tactile surfaces).
- *Appearance*: Verbal, visual, acoustic, and tactile perceptual characteristics of displays. The term implies all aspects of visible, acoustic, and haptic languages (e.g., typography or color; musical timbre, or cultural accent within a spoken language; and surface texture or resistance to force), as well as the level of abstraction or realism in graphic imagery.

Based on knowledge of the users, developing a UI suitable for global deployment may affect each of these components: from choices of metaphorical references to hierarchies in the mental model, from navigation complexity to decisions on appropriate input techniques, and from selection of character sets, graphics, colors, sounds/voice/music to the alternative use of vibration.

Fulfilling the designers' second objective requires an understanding of internationalization, translation, and localization issues within the context of a globalization strategy.

Globalization refers to the entire process of preparing products or services for worldwide production and consumption. Globalization includes analysis of issues at international, intercultural, and local scales. In our information-oriented society, globalization affects almost all types of computer-mediated communication and interaction.

Internationalization refers to the process of preparing code that separates data and resources (i.e., items needed for input and output) from the primary functionality of the software. Software created in this way does not need to be rewritten or recompiled for each local market. This separation may also include the ability for the UI to work on different platforms in one or more geographic region.

Typically, internationalization is required due to geographic, political, linguistic, and typographic differences between nations or groups of nations. For instance, different countries will have different provinces, states, or political subdivisions and different denominations or metrics for currency, time, and physical measurements that need to be accommodated in pull-downs, menus, or fields (Table 15.1). People in global cities like Dubai, and countries with high levels of immigration, may also need to take advantage of internationalization features due to differences in religion, dialects, esthetics, or other humanistic issues of particular importance. Examples include calendars that acknowledge different religious time cycles; terminology for color, type, and signs that reflect different popular cultures; and web search criteria based on cultural preferences. Sometimes these issues reach national political importance, as in the 2005 debate in Iraq over whether Thursday and Friday (the Muslim Sabbath) should constitute the "weekend" rather than including Saturday (the Jewish Sabbath) or Sunday (the Christian Sabbath) (Kuhn 2005).

Technical, financial, political, and legal matters also affect designs. The European Union has developed a variety of requirements for open source and software interoperability that have led to prolonged litigation against Microsoft (Raby 2007); computer programs imported into Canada must meet a legal requirement for bilingual English and French displays; and national requirements for censorship (and privacy) continue to affect the use of Internet browsers and social media in parts of Asia (Helft and Barboza 2010). The ISO has developed a number of software, hardware, and human factors standards in an effort to establish international standards for some parts of a UI.

Translation, the conversion of text from one language to another, can be accomplished by certified in-house staff or by one or more outside contractors specialized in that service. Translators generally use software provided by third-party firms (e.g., Systran), for a preliminary pass, but it is vital that someone skilled in the language review the output.

It is tempting to rely on translation software, but it remains a "work in progress." According to Quebec translator Gisèle Foucault (2010), a better understanding of the current types of systems explains why.

In automatic translation systems, like Systran, the computer performs the translation based on dictionaries, grammar algorithms, and so on. In very specialized areas where the vocabulary is well established, metaphors are minimal, and the system has been "trained" on large volumes of documents revised by human translators, the systems can eventually provide acceptable translations. After considerable effort, the European Commission now uses Systran for administrative documents. In Canada, an automatic translation system has been developed for weather forecasts, which is also quite good. However, such success stories remain limited. Google and others browsers offer such automatic translation systems, but the sheer variety of vocabulary and contexts in which words are used means that translations are sometimes defective. In particular, industry-specific vocabulary may not be included in the application's internal dictionary. As an example, the verb to "spawn" (as in an Internet application that "spawns windows") is defined as "fish eggs" in some Chinese language dictionaries.

Translation memory systems, like the Trados Translator's Workbench or SDLX, are the second category of translation software. A human being does the translation, and the computer saves the original sentence and corresponding translation for future use should a similar sentence (100% match or 70%–100% "fuzzy" match) be found again. Thus, the initial

TABLE 15.1
Examples of Differing Displays for Currency, Time, and Physical Measurements

Item	U.S. Examples	European Examples	Asian Examples
Currency	$1,234.00 (U.S. Dollars)	DM1.234 (German marks)	¥1,234 (Japanese yen)
Time measures	8:00 PM, August 24, 1999	20:00, 24 August 1999 (England)	20:00, 1999.08.24, or Imperial Heisei 11, or H11 (Japan)
	8:00 PM, 8/24/99	20:00, 24.08.99 (Germany, traditional) 20:00, 1999-08-24 (ISO 8601 Euro standard)	
Physical measures	3 lb, 14 oz	3.54 kg, 8.32 m (England)	3.54 kg, 8.32 m in Roman or Katakana
Chars	39 100, 3 feet and 10 inches	3,54 kg, 8,32 m (Euro standard)	

translation can be reused as is, or with minimal revision. However, this kind of system is only as good as the human translator who provides the initial translation (although the translations memorized by the system can always be edited by the next translator). Such systems are very useful for online help files, software user guides, training modules, or any text with much repetition (e.g., contracts).

In Canada, work is progressing on a third type of system, statistical machine translation. It is a blend of the two other types, with statistical analysis of an extremely large corpus of human-translated documents (in the Canadian case, the Hansard, deliberations of the House of Commons). The system analyzes one word, then two words, then three words, and so on, and their equivalents in French. With the help of statistical formulas, the system matches "co-occurrences" with their equivalents and "learns" to translate in context. If the product becomes a commercially viable product, it could change the translation market in specific niches.

Ideally, translations are checked by having them back-translated into the original language. This technique is particularly important for a text that asks users to reply to a prompt or question, and the technique ensures consistency within data collected across linguistic boundaries.

Preparing texts in local languages often requires the use of additional or different characters. The American Standard Code for Information (ASCII) system, which uses seven or eight bits to represent characters, supports English, and the single-byte ISO 8859-1 character set supports Western European languages that use the Latin alphabet, such as Spanish, French, and German. Other character encoding systems include EBCDIC, Shift-JIS, UTF-8, and UTF-16. ISO has established specific character sets for languages such as Cyrillic, Modern Greek, Hebrew, Japanese, and so on. However, most companies planning to globalize their products should use Unicode, a double-byte (16 bit) system that can represent 65,536 characters, which is sufficient to display Asian languages like Japanese and Korean and permits easier translation and presentation of character sets.

Localization may or may not be required depending on the degree of cultural difference between the original and global target users. For instance, a Canadian company selling into the United States generally does relatively little localization to prepare its products for sale. On the other hand, preparing an application for a specific, small-scale community (less than a country) or cross-national, ethnic "region" (unified by language and culture) may require significant localization. That same Canadian company selling to Nunavut (the federal Inuit territory in northern Canada) may make extensive changes to its product. Web-based companies planning on doing business in the European Union know that many language versions will be required for basic commercial penetration into the desired markets. Some situations may benefit from a culture audit, in which the icons, graphics, terminology, and concepts are inspected to alert translators to potential difficulties or dangers when translating. Otherwise, a perfectly good translation into another language and writing system may overlook items that are culturally alienating or offensive, requiring expensive later modifications when the product/service has already become available to users commercially.

Used informally, localization also applies to "corporate cultures" or age-stratified groups, which may be geographically dispersed. For instance, multinational companies with strong socialization practices can develop surprisingly coherent cultures centered on mission statements and "core values." One researcher, Geert Hofstede (1997), took advantage of this uniformity in his study of intercultural differences in 53 countries and regions by studying employees of only one company, IBM.

Other claims are made for the cultural stability of age cohorts like the current "millennials" or "digital natives." Digital natives have grown up surrounded by computers, mobile devices, video games, and the Internet, whereas others who have always used some other form of technology first are termed "digital immigrants," held back by their initial impressions of the "right" (older or previous) way to do things. Many claims have been made about how digital natives differ from digital immigrants, such as the following quote by Prensky (2001a) discussing changes in education:

> Digital Natives are used to receiving information really fast. They like to parallel process and multi-task. They prefer their graphics before their text rather than the opposite. They prefer random access (like hypertext). They function best when networked. They thrive on instant gratification and frequent rewards. They prefer games to "serious" work.… But Digital Immigrants typically have very little appreciation for these new skills that the Natives have acquired and perfected through years of interaction and practice.

In enhancing products for teamwork and coordination, localization might be required to address the different needs of different age cohorts. One could look at digital natives (millenials) versus digital immigrants or look more closely at the differences between millenials and baby boomers (those born between 1946 and 1964), Gen X (those born after the baby boomers, generally from about 1961–1970 until about 1980–1981), Gen Y (those born after Gen X, generally from about 1981 until about 1994–2000. Each of these age cohorts is distinguished by different levels of interest and trust in a variety of technologies.

Finally, localization can include affinity groups (e.g., French "twenty-somethings" or German Mercedes automobile owners), business or social organizations (e.g., U.S. golf club members), or specific intranational groups (e.g., Swedish househusbands, India's untouchables, or young Japanese professional women who are rejecting marriage as a life pattern). With the spread of web access, the term culture is frequently applied to all kinds of groups with strongly shared interests, members of which may or may not be geographically dispersed.

Note: This broad definition of "culture" is not accepted by all theorists (e.g., see Clausen as reported by Yardley 2000). However, for the purposes of this chapter, this broad definition is used.

15.2.2 User-Interface Elements

Ultimately, a globalization strategy may require designers to consider any or all of the following elements of a desktop, website, mobile device, appliance, vehicle display, or other kind of software application:

- Access limitations (e.g., children or the blind)
- Address formats
- Alphabetic sequence and nomenclature
- Arithmetic operations symbolism
- Business standards (quotes, tariffs, contracts, agreement terms, etc.)
- Calendars
- Censorship
- Character handling
- Colors
- Content categories
- Date and time formats
- Distance references
- Documentation nomenclature and formats
- Electrical and electronic plug formats and nomenclature
- Energy formats
- Environmental standards (“green” compliancy, low energy, low pollution, etc.)
- File formats
- Font nomenclature, sizes, faces, and byte formats
- Frequency (e.g., gigahertz)
- Hyphenation and syllabification
- Icons and symbols (e.g., mailboxes, trashcans, files)
- Intellectual property (protection via patents, copyrights, trademarks)
- Keyboard formats
- Language and dialect differences
- Legal processes
- Licensing standards
- Measurement units (length, volume, weight, electricity, energy, temperature, etc.)
- Monetary or currency formats
- Multilingual usage
- Name formats
- Negative formats
- Numeric formats and number symbols
- Packaging
- Paper formats
- Privacy
- Punctuation symbols and usage
- Reading/writing direction
- Sorting sequences
- Style formats
- Telephone/fax standards
- Temperature formats
- Text length (especially, expansion requirements for translation)
- Video recording and playback formats
- Voltage/amperage units and formats
- Weight formats

Table 15.2 hints at the potential complexity of globalizing products for North America/Europe, the Middle East, and East Asia. Table 15.3 shows how references can change even among English-language users.

Many of these user-interface elements are specified as industry standards. Unicode was developed by the Unicode Consortium, whereas other language encoding formats are supported by ISO. A variety of formats, such as calendars, dates, time zone, numbers, currency values, and sorting systems, are listed in the Unicode Consortium’s CLDR (Common Locale Data Repository) charts, available from cldr.unicode.org/.

TABLE 15.2
Examples of Differing Displays for Other Data Formats

Item	U.S. Examples	European Examples	Asian Examples
Numerics	1,234.56 (also Canada, China, United Kingdom)	1 234,56 (Finland, France, Luxembourg, Portugal, Sweden) 1.234,56 (Albania, Argentina, Denmark, Greece, the Netherlands) 1’234.56 (Switzerland: German, Italian) 1’234,56 (Switzerland: French)	1,234.56
Telephone numbers	1-234-567-8901, ext. 23 1.234.567.8901 (123) 456-7890	1234 56 78 90 (Austria) (123) 4 5 6 78 90 (Germany) (12) 3456 789 (Italy) 146(0)12 345 67 149 (1234) 5678-9 (Switzerland)	181-53-478-1481 (Japan) 82 2 3142 1100 (Korea) 182-(0)2-535-3893 (Korea) 86 12 34567890 (China)
Address formats	Title, First Name, MI, Last Name Department Company Number, Street, City, State, Zip-Code, Country	Paternal Name, Maternal Name, First Name Company, Department Street, Number City, District/Region Zip-Code, Country (Order may vary from country to country)	Family Name, First Name Department Company Number, Street, Neighborhood, District, Zip-Code, City (Japan)

Source: Partially from Aykin, N., ed. 2005. Usability and Internationalization of Information Technology. Mahwah, NJ: Lawrence Erlbaum.

TABLE 15.3
Examples of Differing Cultural References

Item	North America/Europe Example	Middle Eastern Example	Asian Example
Sacred colors	White, blue, gold, scarlet (Judeo-Christian)	Green, light blue (Islam)	Saffron yellow (Buddhism)
Reading direction	Left to right	Right to left	Top to bottom
Item	United States	France, Germany	Japan
Web search	"Culture" doesn't imply political discussions	"Culture" implies political discussions	"Culture" implies tea ceremony discussions
Sports	Baseball, football, basketball; golf is a sport	Soccer	Sumo wrestling, baseball; golf is a religion

Additional information on specific internationalization/localization issues can be found through LISA (Localization Industry Standards Association, lisa.org), which has an active program of standards development through OSCAR (Open Standards for Container/content Allowing Reuse).

15.2.3 Globalization Development Processes

The "globalized" UI development process is a sequence of partially overlapping steps, some of which are partially or completely iterative:

- *Plan*: Define the strategy, including the challenges or opportunities for globalization; establish objectives and tactics; and determine budget, schedule, tasks, development team, and other resources. Globalization must be specifically accounted for in each item of project planning; otherwise, cost overruns, delays in schedule, and lack of resources are likely to occur. In most cases, business managers will expect to see a return-on-investment (ROI) analysis of the expected benefits, benchmarking standards, likely tools and process, and metrics to be used in "proving" the results are better.
- *Research*: Investigate dimensions of global variables and techniques for all subsequent steps, for example, techniques for analysis, criteria for evaluation, media for documentation, and so on. In particular, identify items among data and functions that should be targets for change and identify sources of national/cultural/local reference. Globalized user-centered design stresses the need to adequately research users' wants and needs according to a sufficiently varied spectrum of potential users, across specific dimensions of differentiation. In the mid-1980s, Xerox PARC (Palo Alto Research Center Incorporated) began employing cultural anthropologists to study service technicians; by the mid-2000s, Microsoft (and many other multinationals) was hiring anthropologists to undertake ethnographic analyses of work to support product development (Suchman 1987; Murphy 2005). During the past few years, many companies have had anthropologists and ethnographers attend user-interface conferences and anthropology conferences to report on their findings. Several applied anthropology groups regularly post and answer inquiries about information resources, tools, educational institutions, and case studies related to anthropology in design; one of the most active is *anthrodesign* in Yahoo groups.
- *Analyze*: Examine challenges or opportunities in the prospective markets; refine criteria for success in solving problems or exploiting opportunities (write marketing or technical requirements); determine key criteria for usability, usefulness, and appeal (i.e., the user experience); and define the design brief or primary statement of the design's goals. At this stage, globalization targets should be itemized.
- *Design*: Visualize ways to satisfy criteria using alternative prototypes. Based on prior or current evaluations, select the design that best satisfies criteria for both good general UI design and globalization requirements. Prepare documents that enable consistent, efficient, precise, and accurate implementation.
- *Implement*: Build the design to complete the final product, that is, write easily internationalized code using appropriate, effective, efficient tools identified in planning and research steps.
- *Evaluate*: At any stage, review or test results in the marketplace against defined criteria for success, that is, conduct focus groups, test usability on specific functions, and gather sales data and user feedback. Identify and evaluate matches and mismatches, and then revise the designs. Test prototypes or final products with international, intercultural, or specific localized user groups to achieve globalized UI designs. As noted in recent proceedings of UI conferences such as the Association for Computing Machinery's Special Interest Group on Human–Computer Interaction (ACM/SIGHCI), Usability Professionals Association (UPA), Human–Computer Interaction International (HCII), and the International Workshop on the Internationalization of Products and Services (IWIPS), the techniques of evaluation may need to be adjusted to be successful in different cultures and geographic circumstances (Chavan 2005; Gould 2009). Many studies have

focused on website visitors and mobile phone users (Lee et al. 2005; Kim and Lee 2010).

- *Document*: Record the development history, issues, and decisions in specifications, guidelines, and recommendation documents. Honold (1999), for example, notes that German and Chinese mobile phone users require different strategies for documentation, as well as training, which are related to cultural differences predicted by classical models. Even designing the "ideal" documentation for a particular culture of developers may require user-centered design processes, for example, recognizing that some may prefer paper and text, and others visual, interactive means with minimal text reading.
- *Maintain and Train*: Determine which documents, customer-response services, and other processes will require multiple languages, different graphics, and changes in media or delivery techniques. Prepare appropriate guidelines and templates.

15.2.4 Critical Aspects for Globalization: Specific Guidelines

Beyond the high-level UI development process steps identified in the Section 15.2.3, the following guidelines can assist developers in preparing a "checklist" for specific tasks. The following recommendations are grouped under the UI design terms referred to earlier.

15.2.4.1 User Demographics

- Identify national and cultural target user populations and segments within those populations, and then identify possible needs for differentiation of UI components and the probable cost of delivering them.
- Identify potential savings in development time through the reuse of UI components based on common attributes among user groups. For example, certain primary (or top-level) controls in a mobile phone application might be designed for culturally different cognitive styles to aid comprehension and to improve appeal for specific user groups (Kim and Lee 2007). Lower level controls, on the other hand, might be designed with standardized, unvarying form-like elements.
- Consider legal issues in target communities, which may involve issues of religion, privacy, intellectual property, spamming, defamation, pornography and obscenity, vandalism (e.g., viruses), hate speech, fraud, theft, exploitation and abuse (children, environment, elderly, etc.), legal jurisdiction, seller/buyer protection, and so on.

15.2.4.2 Technology

- Determine the appropriate media for the appropriate target user categories, for example, emphasis of sound, visual, or three-dimensional tactile media; verbal versus visual content; and so on.
- Account for international differences in support platform, population and software needs, including languages, scripts, fonts, colors, file formats, and so on.
- Research and obtain appropriate software for code development and content management systems.

15.2.4.3 Metaphors

- Determine optimum minimum number of concepts, terms, and primary images to meet target user needs.
- Check for potential miscommunication and misunderstanding due to differences in language and culture.
- Adjust metaphorical images or text to account for national or cultural differences. Publications and projects from/for China and India have suggested metaphors that differ considerably from Western stereotypes. For example, Chavan (1994) stated that Indians relate more easily to bookshelves, books, chapters, sections, and pages, rather than the desktop, file folders, and files. The Wukong prototype personal digital assistant, developed by Ericsson for Chinese users in 2002, used metaphors based on the Chinese business-social concept of Guang-xi, or relationship maintenance. This approach meant that people, knowledge, and relationships were more fundamental and pervasive than folders, files, and applications (Marcus 2003).

15.2.4.4 Mental Models

- Determine optimum minimum varieties of content organization.
- Consider how hierarchies may need to change in detail and overall in terms of breadth and depth. Chiu (1972) found that United States and Chinese children categorized objects on the basis of different principles: categories and rules by the first, associations in the use of objects by the second. Choong and Salvendy (1999) noted that Chinese and North American users tended to organize the contents of a house in different ways, and Carroll (1999) noted that if one group was given the hierarchies of the other, the group had more difficulty navigating the hierarchy.
- Cognitive styles are also affected by culture. Masuda and Nisbett (2001) found that East Asians and Westerners selected different elements of figure and ground; in their study Asians were more focused on the relationships between elements while Westerners were more focused on central objects. Subsequently, Nisbett and Norenzayan (2002) defined two cognitive perspectives:
  - Holistic thought (orientation to context and the field as a whole, preference for experience-based knowledge, and acceptance of change and multiple perspectives) was linked to personal interdependence and was more common in East Asia.

- Analytic thought (decontextualization of objects, categorical thinking, preference for rule-based logic, and rejection of contradiction) was linked to personal independence and was more common in Europe and North America.

A later study by Knight and Nisbett (2007) found similar cognitive practices within Italy. People in industrialized and relatively independent northern Italy used analytic thought more than people in the more rural and interdependent southern regions. Furthermore, this effect was linked to class; middle-class participants thought more analytically than working class participants. These findings support Hall's (1976) idea that people tend to favor either high- or low-context communication. In high-context communication, messages are nonverbal, indirect, and embedded in the social and physical context of the interaction; in low-context communication, messages are verbal, direct, and stand on their own.

- Investigate different self-concepts and self-representations when considering the development or addition of social media to a product/service. United States market leaders tend to fail when attempting to enter China. Some of their problems are political but other problems stem from national preferences. As of 2010, Renren and QQ had developed strong followings that may be discouraging Facebook from entering the China market. One reason for their success may be different emphases in their applications that reflect different types of social interaction. A recent blog posting on China Hush listed Renren's emphasis on journaling, support for drought victims in southern China, summaries of trends and popular topics, and the ability to see who sees you as more desirable attributes than Facebook's simple lists and privacy functions; the author (not Chinese) was both supported for recognizing Chinese differences and condemned for simplistic thinking (China Hush 2010).

15.2.4.5 Navigation

- Determine need for navigation variations to meet target user requirements, determine cost-benefit, and revise as feasible. Studies by Dong (2007) showed significant differences of navigation among Chinese, Korean, and U.S. viewers of web pages presented with the same layouts but different scripts and languages. Asian viewers tended to look around the page more in a circular pattern looking at the relationships of items, while U.S. viewers looked more quickly in a figure S path sequentially through major "monuments," then clicked for further information. These differences were studied using typical eye-tracking equipment used for website analysis.

15.2.4.6 Interaction

- Determine optimum minimum variations of input and feedback. For example, because of web-access speed differences for users in countries with very slow access, it is important to provide text-only versions, without extensive graphics, as well as alternative text labels to avoid graphics that take considerable time to appear.

15.2.4.7 Appearance

- Determine optimum minimum variations of visual and verbal attributes. Visual attributes include layout, icons, and symbols, choice of graphics, typography, color, and general esthetics. Verbal attributes include language, formats, and ordering sequences. For example, many Asian written languages, such as Chinese and Japanese, contain symbols with many small strokes. This factor seems to lead to an acceptance of higher visual density of marks in complex public-information displays than is typical for Western countries. In the mobile device realm, Letowt-Vorbeck (2010) citing user testing in South Africa, note that some users in developing companies may be partially or completely illiterate, causing significant challenges in designing visual, nonverbal assistance in the UI.

15.2.5 Example of Specific Guidelines: Appearance

Guidelines for visual and verbal appearance follow. Further details can be found in (Aykin 2005; Gould 2001; Marcus and Gould 2000; del Galdo and Neilson 1996; Fernandes 1995; Nielsen 1990).

15.2.5.1 Layout and Orientation

- Adjust layout of menus, tables, dialogue boxes, and windows to account for varying reading directions and size of text. Roman languages read only left to right, but Asian languages may read in several directions. For example, in Japan, people read from the right and down, or down and to the right, and Arabic/Hebrew may include right-reading Roman-letters text within left-reading lines of Arabic/Hebrew text.
- If dialogues use sentence-like structure with embedded data fields and/or controls, these areas may need special restructuring to account for language changes that significantly alter sentence format. For example, German sentences often have verbs at the ends of sentences, while English and French verbs are placed in the middle of sentences.
- As appropriate, change layout or imagery that implies or requires a specific reading direction. Left-to-right sequencing may be inappropriate or confusing for use with right-to-left reading scripts and languages. A website design for Arabia online

of the late 1990s featured left-to-right English text on a home page intended for Western business people and tourists, but the page layout was still right-to-left, as in Arabic, with primary links at the far right, secondary, and tertiary links in the center and left sections of the screen, and small directional arrows pointing right to left. The designers had retained Arabic-language influence to the detriment of English-language readers.

- Check for misleading arrangements of images that lead the viewer's eye in directions inconsistent with language reading directions.
- For references to paper and printing, use appropriate printing formats and sizes. For example, in the United States, standard office letterhead size is 8.5 by 11 inches, but in Europe it is 210 by 297 mm.

15.2.5.2 Icons, Symbols, and Graphics

- Avoid the use of text elements and punctuation within icons and symbols to minimize the need for versions to accommodate varying languages and scripts.
- Adjust appearance and orientation to account for national or cultural differences. For instance, office equipment such as telephones, mailboxes, folders, and storage devices differs significantly from nation to nation. Using a mailbox as an icon for e-mail may require different images for different countries. A U.S. mailbox looks quite different from a British postbox. It may also be necessary to adjust visual references to technology that many younger users could find confusing. For example, many software products continue to use the image of a small "floppy disk" to represent saving a file to media other than the computer's hard drive. As Patrick Hofmann, a visual designer from Google pointed out at the UPA 2010 conference (Hoffmann 2010), regarding icon recognition worldwide, many younger users have never used or seen a floppy disk of the 1980s or early 1990s; some think it looks like a garage with a door, but come to accept this seemingly strange visual symbol for "save to media."
- Consider using signs or derivatives from international signage systems developed for safety, mass transit, and communication (American Institute of Graphic Arts 1981; Olgyay 1995; Pierce 1996). These signs require little or no translation and may require minimal culture-specific support information due to their use in other contexts.
- Avoid puns and local, unique references that will not transfer well. In addition, note that many "universal" signs may be covered by international trademark and copyright use, for example, Mickey Mouse and the "Smiley" smiling face. Similarly, the humor of "a little old lady in tennis shoes" is lost on people who view age and femininity in other ways.
- Check for appropriateness and use the following with caution:

Animals	People
Body parts/positions	Clothing
Colors	National emblems
Hand gestures	Religious, mythological signs

- Consider whether selection symbols, such as the X or check marks, convey correct distinctions of selected versus not selected items. Some users may interpret an X as crossing out what is not desired, not as selection.
- Symbols of merit or trustworthiness can vary. In a Western auction site like eBay.com, a high-rated vendor is identified by a "seal of approval." By contrast, the Chinese auction site Taobao uses an icon of two hands cradling a golden shopping bag that is often described as a "flower."

15.2.5.3 Typography

- Recognize that character-coding schemes often differ dramatically for different languages. ASCII is primarily limited to English, single-byte schemes accommodate European languages, and double-byte Unicode best supports Asian languages. These differences, as well as bidirectional fonts for Hebrew and Arabic display, make it more challenging to support multilingual UIs. Unless developers can switch (or allow users to switch) character-coding schemes, it is difficult for users to access content easily.
- Use fonts supported for the range of languages required.
- Consider whether special font characters are required for currency, physical measurements, and so on.
- Use appropriate alphabetic sequence and nomenclature (e.g., U.S. "zee" vs. Canadian/English "zed").
- Ensure appropriate decimal, ordinal, and currency number usage. Formats and positioning of special symbols vary from language to language.
- Consider appropriate numeric formats for decimal numbers and their separators (Aykin 2000):

1,234.56	Canada, China, United Kingdom, United States
1.234,56	Albania, Argentina, Denmark, Greece, the Netherlands
1 234,56	Finland, France, Luxembourg, Portugal, Sweden
1’234.56	Switzerland (German, Italian)
1’234,56	Switzerland (French)

- Other numeric issues include the following:
  - Names of characters
  - Standards for display of negative and positive numbers
  - Percent indication

    - Use of leading zeros for decimal values (e.g., 0.1 or .1)
    - List separators
    - Lucky and unlucky numbers (e.g., "lucky" telephone numbers in Asian countries sometimes sell for higher prices. Note that Chinese prefer even numbers, but Japanese prefer odd numbers.)
- Use appropriate temperature formats, that is, Fahrenheit, Centigrade, and Kelvin.
- Use appropriate typography and language for calendar, time zone, and telephone/fax references.
- Consider these date and time issues, among others:
    - Calendars: Gregorian, Muslim, Jewish, Indian, Chinese, Japanese, and so on. Note that within India there are three major Hindu religious calendars (one used by northern Indians, one by southern Indians, and one shared by both) separate from a secular calendar
    - Character representation: Hindu-Arabic, Arabic, Chinese, Roman, and so on
    - Clock of 12 or 24 hours
    - Capitalization rules: for example, book titles in Britain capitalize the first word only versus all main words in U.S. titles and all nouns in German
    - Days considered for start of week and for weekend: Christian, Muslim, and Israeli calendars have different work and rest days. North American monthly calendars typically begin on Sunday; European monthly calendars typically begin on Monday
    - Format field separators
    - Maximum and minimum lengths of date and time
    - Names and abbreviations for days of week and months: two-, three-, and multicharacter standards
    - Short and long date formats for dates and times
    - Time zone(s) appropriate for a country and their names
    - Use of AM and PM character strings
    - Use of daylight savings time
    - Use of leading zeros
- Consider monetary format issues:
    - Credit-/debit-card formats, usage conventions
    - Currency names, denominations
    - Currency symbols (local vs. international versions)
    - Currency conversion rates
    - Monetary formats, symbols, and names
    - Rules for combining different monetary formats: for instance, Euros are often listed with local currencies
    - Requirements for validating monetary input
- Consider name and address formats:
    - Address elements and number of lines
    - Address line order
    - Address punctuation and listing of street numbers (first or last): for example, 4, route de Monastère; 1504 South 58th Street; Motza Illit 11
    - Address zip/postal codes: numeric versus alphanumeric, typography, order in relation to city, state/province, or country
    - Character sets
    - Data field labels (family name vs. last name vs. surname; first name vs. given name vs. Christian name)
    - Field labels: city/town/district/province, and so on
    - Location and location order: neighborhood, district, city/town, state/province
    - Name formats: name order (e.g., family name first for Asian names); capitalization (family names in all caps in Asia); number of names (one name is common in Indonesia); number and order of names (even within Spanish-speaking countries, some list double family names with maternal first, and others list family names with paternal first); surnames (maternal vs. paternal); suffixes (e.g., Jr.); use of middle name or initials
    - Name honorifics, prefixes, and titles: use of Mrs, Miss, or Ms. in English-speaking countries; German double titles like Dr. Eng.
- Consider telephone, fax, and mobile phone number formats:
    - Grouping of digits varies from country to country
    - Internal dialing (initial area code zeros) versus external (without)
    - Numeric versus alphanumeric (i.e., 11-510-767-2676 vs. 510-POP-CORN)
    - Number grouping, separators (e.g., comma, hyphen, period, space, etc.)
    - Use of plus sign for country codes in telephone numbers
    - Use of parentheses for area codes
        – Format for multiple sequential numbers of businesses (slash, commas, etc.)

15.2.5.4 Color

- Follow perceptual guidelines for good color usage. For example, use warm colors for advancing elements and cool colors for receding elements; avoid requiring users to recall in short-term memory more than five plus-or-minus two different coded colors.
- Follow appropriate professional/popular usage of colors, color names, denotation, and connotation.
- Respect national and cultural variations in colors, where feasible, for the target users. For example, Kim and Moon (1998, pp. 16–17) tested emotional

responses to early banking interfaces In Korea. They found color was extremely important in enhancing (or diminishing) trustworthiness:

> The color layout of the interface is also apparently important in enhancing the extent of trustworthiness which the customer of the cyber-banking system feels.… The preferable tone of color for the interface should be cool rather than/ warm and its main color should be a moderate pastel color. At the same time, the colors used in the interface should be of low brightness and the colors should be used symmetrically. On the other hand, the interface that has a bright color background and uses an asymmetrical color scheme will induce a feeling of untrustworthiness in the cyber-banking system.

- In other countries, strong colors and layouts may be used to evoke professionalism and trustworthiness.
- Aykin and Milewski (2005) list color meanings in various countries. For instance, yellow is the imperial color in China but seen as a color of mourning in Egypt.

15.2.5.5 Esthetics

- Respect, where feasible, different esthetic values among target users. For example, some cultures have significant attachment to wooded natural scenes, textures, patterns, and imagery (e.g., the Finnish—Figure 15.1—and the Japanese). Chinese, Korean, and Japanese people enjoy cartoon characters appearing in their products and services, more than would seem appropriate for other cultures.
- Consider specific culture-dependent attitudes. For example, Japanese viewers find disembodied body parts, such as eyes and mouths, unappealing in visual imagery.
- Some cultures prefer to adopt designs from market-leading software. A recent study by Marcus's firm looked at localization issues for Saudi Arabia; Microsoft icons were preferred due to the company's world renown. Similarly, Gould (2007) found that aesthetic standards of "professional design" were associated with websites developed by well-known global companies. Stille (2002) notes that copying is itself a cultural characteristic. Figures 15.2 and 15.3 show the similarity between Baidu.com, a leading Chinese search engine, and Western market leader, Google.com. (For more on these two companies, see Barboza 2006).

15.2.5.6 Language and Verbal Style

- Consider which languages are appropriate for the target users, including the possibility of multiple national languages within one country. For example, English and French within Canada; French, German, and Italian within Switzerland; French or Dutch in Belgium; and Hebrew, Arabic, French (official), and English (unofficial) in Israel. South Africa has eleven official languages (with English listed sixth); India has more than 20. Note also that some languages have different dialects, for example, Mexican, Argentinian, and Castillian (Spain) Spanish; Parisian, Swiss, and Canadian French.

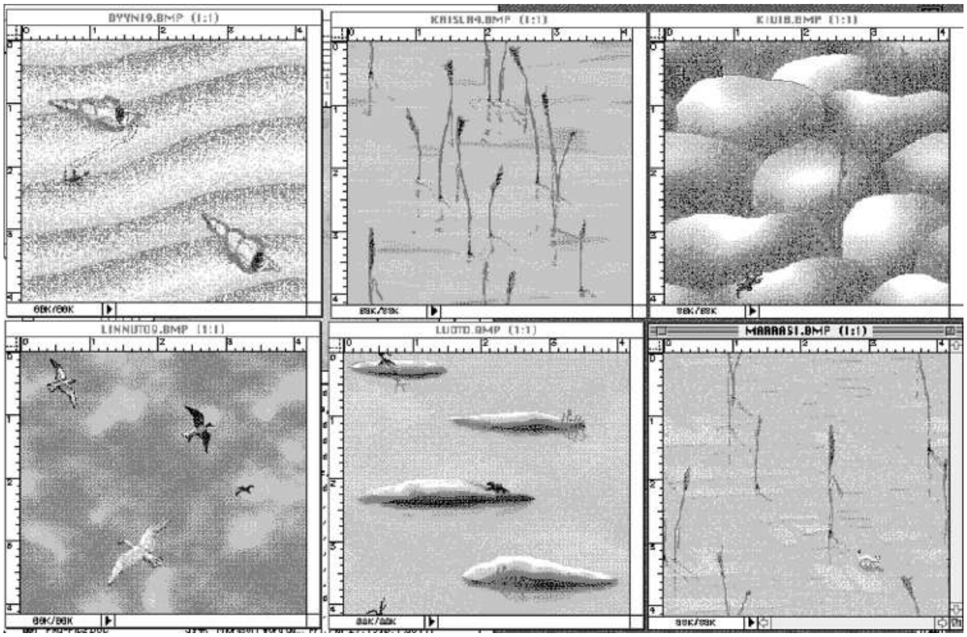


FIGURE 15.1 Example of window application background patterns that seemed appropriate to Finnish software developers. This imagery might be suitable for Finnish and perhaps Japanese, but might not be ideally suited for other cultures.

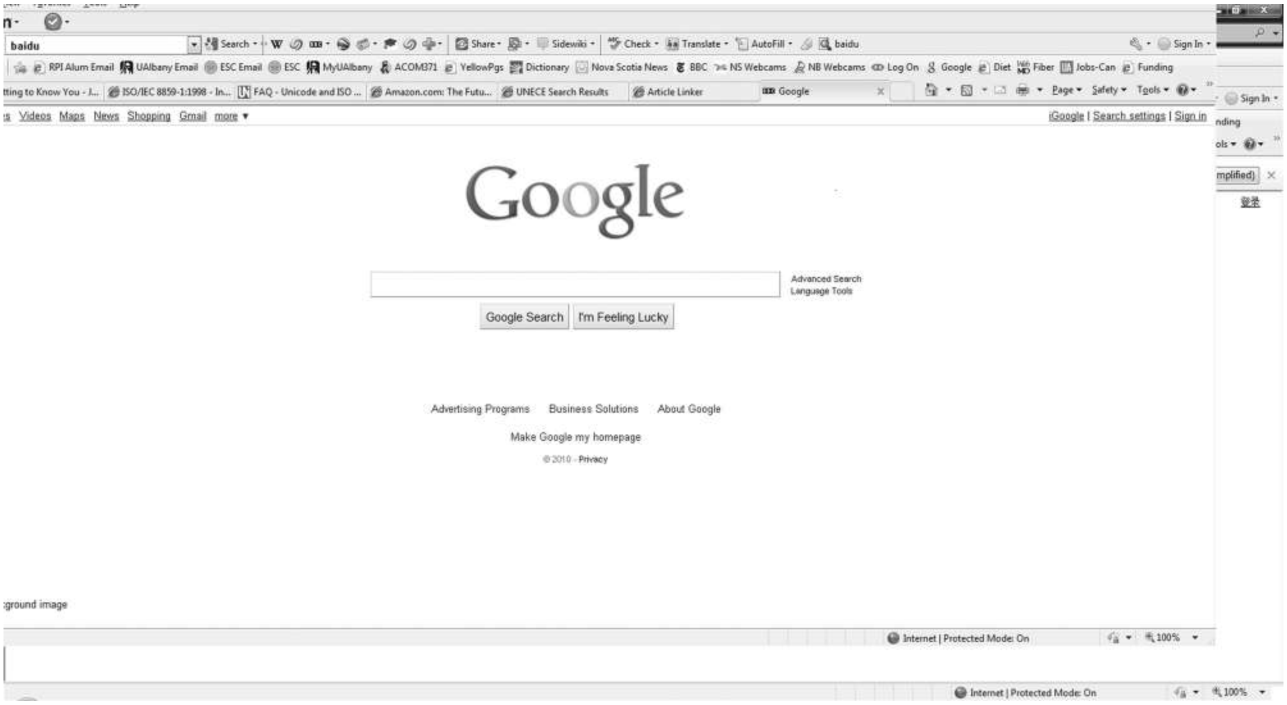


FIGURE 15.2 Example of a Baidu.com Chinese search engine screen home page, which bore a strong resemblance to Google's home page.

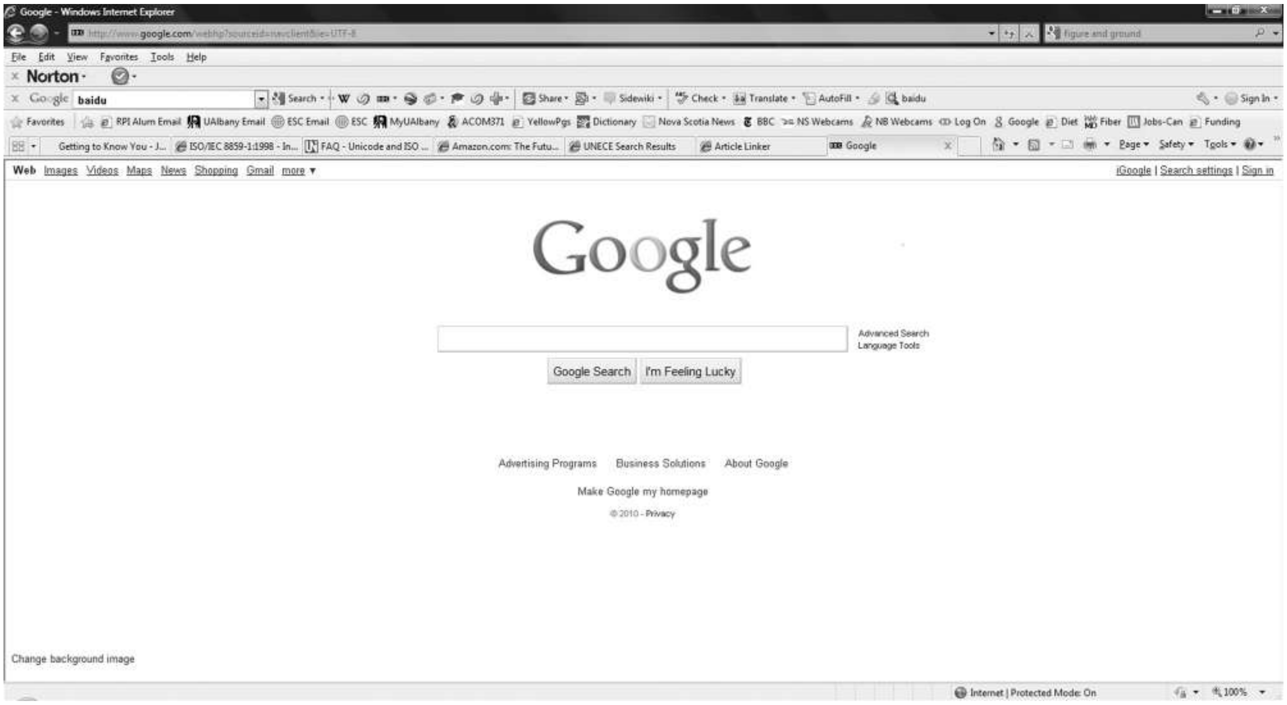

Google - Windows Internet Explorer
http://www.google.com/webhp?sourceid=navclient&ie=UTF-8
File Edit View Favorites Tools Help
Norton
baidu
Web Images Videos Maps News Shopping Gmail more ▾
iGoogle | Search settings | Sign in
Google
Advanced Search
Language Tools
Google Search
I'm Feeling Lucky
Advertising Programs
Business Solutions
About Google
Make Google my homepage
©2010 - Privacy
Change background image
Internet | Protected Mode: On
100%

FIGURE 15.3 Example of Google.com search engine screen home page. This particular design was noted for its simplicity and large amount of white space, which establish a strong brand and visual identity.

- Understand which dialects and variant spellings are appropriate within language groupings and check vocabulary carefully: British versus American terms in English (see Table 15.4), Mexican versus Spanish terms in Spanish, or Mainland China versus Taiwanese terms in Mandarin Chinese. For example, the use of the word *cheque* tends to identify a website as coming from a British or Commonwealth country.
- Analyze the impact of varying languages on the length and layout of text. For example, German,

TABLE 15.4
Comparison of English-Language User Community Conventions

	United States	United Kingdom
Dates	Month/Day/Year: March 17, 2001, 3/17/01	Day/Month/Year: 17 March 2001, 17/03/01
Time	12-hour clock, AM/PM No leading zero (8:32 AM)	24-hour clock Leading zero (08:32)
Currency	US$189.56, 56¢	GB£189.56, £189.56, 56p
Spelling	Center Color	Centre Colour
Terminology	Truck Bathroom	Lorrie Toilet
Book spine title	Top-down	Bottom up

French, and English versions of text generally have increasingly shorter lengths. Some Asian texts are 50%–80% shorter than English; some non-English Roman-character prose texts can be 50%–200% longer. Some labels can be even longer.

Example (Aykin 2000):

English	Undo	Dutch	Ongedaan maken
English	Autoscroll	Swedish	Automatisk rullning
English	Preferences	German	Bildschirmeinstellungen

- Kwintessential.co.uk (2010) provides some examples of expansion/contraction requirements for various pairs of languages.
- Accommodate the different alphabetic sorting or ordering sequences for the varied languages and scripts by preparing variations that correspond to the alphabets. Different languages may place the same letters in different locations; for example, Å comes after A in French but after Z in Finnish.
- Consider differences of hyphenation, insertion point location, and emphasis, for example, use of bold, italic, quotes, double quotes, brackets, and so on.
- Use appropriate abbreviations for typical items such as dates, time, and physical measurements.

15.3 CULTURE AND LOCALIZATION OPPORTUNITIES

15.3.1 Definitions of Culture

Localization goes beyond software internationalization and language translation to consider target market cultures. Culture has many definitions, which can elicit intense discussion. However, for the purposes of global UI design, it is important to recognize that cultural differences are real. This section analyzes some of the needs, wants, preferences, and expectations of different cultures. Most of the analysis is based on a cross-cultural theory developed by Geert Hofstede; however, other competing theories exist, a discussion of which follows.

As an example of culture differences, consider the order in which one might wish to customize a mobile phone. Some people find it most intuitive to work logically: begin on the main screen by selecting Settings and add sound and display preferences. Others find it easier to work thematically: select Sound (and later Display) on the main screen, review possible options, and then set one's ringtone and wallpaper. Is one path better than another? As previously mentioned, it might depend on whether one is an analytical or holistic thinker. Different people, and different cultures, have different cognitive styles and follow different action pathways (Kim and Lee 2007).

Kim and Lee (2010) note that most academic views of culture rely on similar models of perception and cognition and then contrasts models by Bosa, White, Spradley, Hall, Trompenaars, Hoft, Vask and Grantham, Kluckhohn, and Stewart and Bennett. Most such models consist of layers (or levels) that move from unconscious sensation to conscious awareness. In terms of technology, the top layer is the artifact which reflects the values of designers and users, and basic assumptions from their (generally) shared culture. Lee cites Doblin who said that, the product is "frozen information." Adler (2002) similarly notes that cultural symbols and behaviors rest on attitudes and values and can be used to uncover cultural orientations.

However, one of the first issues when dealing with culture is to recognize that people vary widely; culture itself is not deterministic of what any given person will expect or need from a UI. Even within relatively homogeneous groups, there is individual variation; culture merely reflects the central tendency or consensus of the group. People within single countries may be very different, for example, French-Canadians within Canada, or Muslim groups within European countries. However, even if culture is not always predictive, it is frequently highly descriptive in identifying significant social factors affecting user expectations and use. As such, it can also be viewed as a powerful heuristic for all UI designs. By considering alternative approaches and solutions to interface tasks and problems, an understanding of culture can help designers create much more robust designs that can be more easily localized for new markets.

Many organizational and intercultural communication scholars have published classic studies and theories of culture; in the past decade, culture-oriented works have become better known within the UI design community. Theorists include Hall (1969), Victor (1992), Hofstede (1997), Trompenaars and Hampden-Turner (1998), and Schwartz (2004). Gould (2005) summarizes much of the cultural values literature and suggests implications for UI design. Other researchers have applied these theories to a wide range of social contexts, from business relations and commerce to health and education (e.g., Samovar, Porter, and McDaniel 2008; Adler 2002). Books on "cultural intelligence" promise "the new secret to success."

Recent publications have generated controversy regarding the permanence and identity of many characterizations of cultures. Popular entertainment and the wide availability of new media seem to be washing away cultural differences. The permanence of cultural attributes has been questioned in articles by Ona and Spindle (2000) and Herskovitz (2000), who note the rise of individualism (IDV) in classically collectivist Japan and the acceptance, as well as the influence, of Japanese pop-cultural artifacts (i.e., music, movies, and television) in Asian nations that were recently mortal enemies of Japan. Friedman (2005) notes that the world is "flat" (also "hot and crowded," 2008). However, during the same period, Nisbett (2003) was conducting his research program on analytic and holistic thought, demonstrating differences that he attributes to the origins of Eastern and Western cultures in Confucian China and classical Greece.

Other scholars have criticized technology and popular culture as real manifestations of culture. Clausen (2000), as reported by Yardley (2000), argued that the anthropological term "culture" refers to "the (essentially inescapable) total structure of life of a particular society" while many today use culture to refer to (optional) "shared values." By contrast, "…the 'culture' of the Internet has none of the characteristics of a real culture. It is not a total way of life; it did not evolve among a distinct people; nobody inherited it or was raised in it; it makes no moral demands, has no religion at its center, and produces no art." Although complex, its rules are procedural. From the perspective of Internet "dwellers," some of his statements are likely to stir debate, as will his assertions that in terms of the strict definition, culture no longer exists in the United States. At the very least, Clausen stands in direct contrast to Johnston and Johal (1999) who early defined the Internet as a "virtual cultural region" inhabited by low power distance (PD), individualistic, risk-taking men (and a few women).

Hofstede (1997) has discussed the relation of culture to economic success and made notably differing assertions. He feels people are influenced by a complex mixture of culture plus geography and idiosyncratic drifts in technology, that is, both culture and creativity. Because we are socialized as children, people tend to retain their cultural orientations, values, and attitudes throughout their lives. Self-image may be undermined by change but, even in denial, people remain molded by their culture.

The application of Hofstede's theories will demonstrate the value of intercultural communication research for UI design, using examples in web design. The following Sections 15.3.2 to 15.3.2.5 introduce Hofstede's concept of cultural dimensions.

15.3.2 Hofstede's Dimensions of Culture

During 1978–1983, the Dutch cultural anthropologist Geert Hofstede conducted detailed interviews and surveys with thousands of IBM employees in 53 countries. Through standard statistical analysis of large data sets, he was able to determine patterns of similarities and differences among their replies. From this analysis, he formulated his theory that world cultures vary along consistent, fundamental dimensions. Because his subjects were constrained to one multinational corporation's worldwide employees, and thus to one company culture, he ascribed their differences to the effects of their national cultures. (One debated characteristic of his approach is that he maintained that each country has just one dominant culture.) Hofstede's methodology has been frequently criticized (McSweeney 2002) but his work has tremendous face validity and is widely used in the fields of management and intercultural and organizational communication.

In 1997, Hofstede published a version of his research as *Cultures and organizations: Software of the mind*. His focus was not on defining culture as refinement of the mind but rather on essential patterns of thinking, feeling, and acting that are well established during childhood. These cultural differences manifest themselves in a culture's choices of symbols, heroes/heroines, rituals, and values.

Hofstede rated 53 countries on indices including five dimensions normalized to values (usually) of 0–100. His five dimensions (indices) of culture are as follows:

- Power distance (PDI)
- Collectivism/individualism (IDV)
- Femininity/masculinity (MAS)
- Uncertainty avoidance (UAI)
- Long-term/short-term time orientation (LTO)

Each of Hofstede's dimensions follows with an explanation of implications for UI (especially web UI) design, and illustrations of characteristic websites. The complete data for all countries appears in Table 15.5.

15.3.2.1 Power Distance

PD refers to the extent to which less powerful members expect and accept unequal power distribution within a culture.

High-PD countries tend to have centralized political power and exhibit tall hierarchies in organizations with large differences in salary and status. Subordinates may view the boss as a benevolent dictator who expects them to do as they are told. Parents teach children to be obedient and expect respect. Teachers are considered wise and are esteemed. Inequalities are expected, and even may be desired.

Low-PD countries tend to view subordinates and supervisors more as equals and more interchangeable, with flatter hierarchies in organizations and less difference in salaries and status. Parents and children, as well as teachers and students, may view themselves more as equals. Equality is expected and generally desired.

Hofstede noted that these differences are hundreds or even thousands of years old. He does not believe they will disappear quickly from traditional cultures, even with powerful global telecommunication systems. Based on this definition, (high vs. low) PD may influence the following aspects of UI design:

- Access to information: highly structured versus not as highly structured
- Hierarchies in mental models: tall versus shallow

TABLE 15.5
Indices from Hofstede

	PDI Rank	Score	IDV Rank	Score	MAS Rank	Score	UAI Rank	Score	LTO Rank	Score
Arab countries	7	80	26/27	38	23	53	27	68		
Argentina	35/36	49	22/23	46	20/21	56	10/15	86		
Australia	41	36	2	90	16	61	37	51	15	31
Austria	53	11	18	55	2	79	24/25	70		
Bangladesh									11	40
Belgium	20	65	8	75	22	54	5/6	94		
Brazil	14	69	26/27	38	27	49	21/22	76	6	65
Canada	39	39	4/5	80	24	52	41/42	48	20	23
Chile	24/25	63	38	23	46	28	10/15	86		
China									1	118
Columbia	17	67	49	13	11/12	64	20	80		
Costa Rica	42/44	35	46	15	48/49	21	10/15	86		
Denmark	51	18	9	74	50	16	51	23		
East Africa	21/23	64	33/35	27	39	41	36	52		
Equador	8/9	78	52	8	13/14	63	28	67		
Finland	46	33	17	63	47	26	31/32	59		
France	15/16	68	10/11	71	35/36	43	10/15	86		
Germany FR	42/44	35	15	67	9/10	66	29	65	14	31
Great Britain	42/44	35	3	89	9/10	66	47/48	35	18	25
Greece	27/28	60	30	35	18/19	57	1	112		
Guatemala	2/3	95	53	6	43	37	3	101		
Hong Kong	15/16	68	37	25	18/19	57	49/50	29	2	96
India	10/11	77	21	48	20/21	56	45	40	7	61
Indonesia	8/9	78	47/48	14	30/31	46	41/42	48		
Iran	29/30	58	24	41	35/36	43	31/32	59		
Israel	52	13	19	54	29	47	19	81		
Italy	34	50	7	76	4/5	70	23	75		
Jamaica	37	45	25	39	7/8	68	52	13		
Japan	33	54	22/23	46	1	95	7	92	4	80
Malaysia	1	104	36	26	25/26	50	46	36		
Mexico	5/6	81	32	30	6	69	18	82		
The Netherlands	40	38	4/5	80	51	14	35	53	10	44
New Zealand	50	22	6	79	17	58	39/40	49	16	30
Nigeria									22	16
Norway	47/48	31	13	69	52	8	38	50		
Pakistan	32	55	47/48	14	25/26	50	24/25	70	23	0
Panama	2/3	95	51	11	34	44	10/15	86		
Peru	21/23	64	45	16	37/38	42	9	87		

Philippines	4	94	31	32	11/12	64	44	44	21	19
Poland									13	32
Portugal	24/25	63	33/35	27	45	31	2	104		
Salvador	18/19	66	42	19	40	40	5/6	94		
Republic of Ireland	49	28	12	70	7/8	68	47/48	35		
Singapore	13	74	39/41	20	28	48	53	8	9	48
South Africa	35/36	49	16	65	13/14	63	39/40	49		
South Korea	27/28	60	43	18	41	39	16/17	85	5	75
Spain	31	57	20	51	37/38	42	10/15	86		
Sweden	47/48	31	10/11	71	53	5	49/50	29	12	33
Switzerland	45	34	14	68	4/5	70	33	58		
Taiwan	29/30	58	44	17	32/33	45	26	69	3	87
Thailand	21/23	64	39/41	20	44	34	30	64	8	56
Turkey	18/19	66	28	37	32/3	45	16/17	85		
Uruguay	26	61	29	36	42	38	4	100		
USA	38	40	1	91	15	62	43	46	17	29
Venezuela	5/6	81	50	12	3	73	21/22	76		
West Africa	10/11	77	39/41	20	30/31	46	34	54		
Yugoslavia	12	76	33/35	27	48/49	21	8	88		
Zimbabwe									19	25

Source: Hofstede, G. *Cultures and Organizations: Software of the Mind, Intercultural Cooperation and Its Importance for Survival*, McGraw-Hill, New York, 1997. With permission.

Abbreviations: PDI = power distance; IDV = collectivism/individualism; MAS = femininity/masculinity; UAI = uncertainty avoidance; LTO = long-/short-term time orientation.

- Emphasis on the social and moral order (e.g., nationalism or religion) and its symbols: significant/frequent versus minor/infrequent use
- Focus on expertise, authority, certifications, and official logos: strong versus weak
- Prominence given to leaders versus citizens, customers, or employees
- Importance of security, restrictions, or barriers to access: explicit, enforced, frequent restrictions on users versus transparent, integrated, implicit freedom to roam
- Social roles used to organize information (e.g., a managers' section that is obvious to all but sealed off from nonmanagers)
- Acceptance of website censorship; reduced concern with privacy within social media

15.3.2.2 Individualism versus Collectivism

Individualism in cultures implies loose ties; everyone is expected to look after one's self or immediate family but no one else. *Collectivism* (low IDV) implies that people are integrated from birth into strong, cohesive groups that protect them in exchange for unquestioning loyalty.

Hofstede found individualistic cultures value personal time, freedom, challenge, and such extrinsic motivators as material rewards at work. In family relations, they value being honest, talking things out, using guilt to achieve behavioral goals, and maintaining self-respect. Their societies and governments place individual social-economic interests over the group, maintain strong rights to privacy, nurture strong private opinions (expected from everyone), restrain the power of the state in the economy, emphasize the political power of voters, maintain strong freedom of the press, and profess the ideologies of self-actualization, self-realization, self-government, and freedom.

At work, collectivist cultures value training, physical conditions, skills, and the intrinsic rewards of mastery. In family relations, they value harmony more than honesty/truth (and silence more than speech), use shame to achieve behavioral goals, and strive to maintain face. Their societies and governments place collective social-economic interests over the individual, may invade private life and regulate opinions, favor laws and rights for groups over individuals, dominate the economy, control the press, and profess the ideologies of harmony, consensus, and equality.

Individualism and collectivism may influence, respectively, the following web UI aspects:

- Motivation based on personal achievement: maximized (expect the extraordinary) for individualist cultures versus underplayed (noncompetitive and internal) for collectivist cultures.
- Images of success: demonstrated through materialism and consumerism versus achievement of social, religious, or charitable agendas.
- Rhetorical style: direct, controversial/argumentative speech and tolerance or encouragement of extreme claims versus indirect (apparently vague) speech, official slogans, harmony, modesty, and subdued controversy.
- Prominence given to youth and action versus to aged, experienced, wise leaders and contemplative states of being.
- Importance of individuals; products shown in use by single person versus products shown by themselves or used by groups.
- Central area of focus versus contextual images that place objects against a background. (You [2009] has investigated the relationships of objects to context.)
- Underlying sense of social morality: emphasis on rules and absolute truth versus emphasis on relationships and an ethic of care.
- Emphasis on change: what is new and unique versus what constitutes tradition and maintains historical trends.
- Willingness to provide personal information to all versus sharply defined in-/out-groups (sharing of information with the in-group and protection of personal data from the out-group).

15.3.2.3 Masculinity versus Femininity

Masculinity and *Femininity* (low MAS) refer to gender roles, not physical characteristics. Hofstede focused on the traditional assignment to masculine roles of assertiveness, competition, and toughness, and the assignment of feminine roles to home and children, people, and tenderness. He acknowledged that, in different cultures, different professions are dominated by different genders. (e.g., women dominated the medical profession in the former Soviet Union, while men dominated in the United States). However, in masculine cultures, the traditional distinctions are strongly maintained; whereas feminine cultures tend to collapse the distinctions and overlap gender roles (both men and women can exhibit modesty, tenderness, and a concern with both quality of life and material success). Traditional masculine work goals include earnings, recognition, advancement, and challenge. Traditional feminine work goals include good relations with supervisors, peers, and subordinates; good living and working conditions; and employment security.

The following list shows some typical MAS index values, where a high value implies a strongly masculine culture:

95	Japan
79	Austria
63	South Africa
62	United States
53	Arab countries
47	Israel
43	France
39	South Korea
05	Sweden

Since Hofstede's definition focused on the balance between roles and relationships, masculinity and femininity may be expressed on the web through different emphases. High-MAS cultures might focus on the following UI design elements:

- Traditional gender/family/age distinctions; clothing and personal appearance of men and women expected to be strongly differentiated
- Focus given to masculine accomplishment
- Work tasks, roles, and mastery, with quick results for limited tasks
- Navigation oriented to exploration and control
- Attention gained through games and competitions; games have winners and losers
- Graphics, sound, and animation used for utilitarian purposes

Feminine cultures might emphasize the following UI design elements:

- Blurring of gender roles; clothing and personal appearance of men and women often androgynous
- Mutual cooperation, exchange, and support (vs. mastery and winning)
- Attention gained through visual esthetics, appeals to unifying values, poetry, images of nature

15.3.2.4 Uncertainty Avoidance

People vary in the extent to which they feel anxiety about uncertain or unknown matters, as opposed to the more specific feeling of fear caused by known or understood threats. Cultures vary in their avoidance of uncertainty, creating different rituals and having different values regarding formality, punctuality, legal-religious-social requirements, and tolerance for ambiguity.

Hofstede noted that cultures with high UA tend to have high rates of suicide, alcoholism, and accidental deaths, and high numbers of prisoners per capita. Businesses may have more formal rules, require longer career commitments, and focus on tactical operations rather than strategy. At least in Europe and South Asia, these cultures tend to be expressive; people talk with their hands, raise their voices, and show emotions. People seem active, emotional, and even aggressive. They shun ambiguous situations and expect structure in organizations, institutions, and relationships to help make events easy to interpret and predictable. Teachers are expected to be experts who know the answers and may speak in cryptic, academic language that excludes novices. In high-UA cultures, what is different may be viewed as a threat, and what is "dirty" (unconventional) is often equated with what is dangerous.

By contrast, low-UA cultures tend to be less expressive and less openly anxious; people behave quietly without showing aggression or strong emotions (though their excessive caffeine consumption may be to combat depression from their inability to express their feelings). People seem easygoing, even relaxed. Teachers may not know all the answers (or there may be more than one correct answer), conduct more open-ended classes, and are expected to speak in plain language. In these cultures, what is different may be viewed as simply curious, or perhaps ridiculous.

Based on this definition, UA may influence contrary aspects of UI and web design. High-UA cultures might emphasize the following:

- Simplicity, with clear metaphors, limited choices, and restricted data; high-UI websites would give users far fewer choices to help them avoid making errors
- Attempts to reveal or forecast results of actions before users act
- Navigation schemes intended to prevent users from becoming lost: use of "bread crumbs" and other methods help keep users situated
- Mental models and help systems that focus on reducing "user errors"
- Redundant cues (color, typography, sound, etc.) to reduce ambiguity; lower information density

Low-UA cultures might emphasize the reverse:

- Complexity with maximal content and choices; low-UA websites will give power users much more flexibility
- Acceptance (even encouragement) of wandering and risk, with a stigma on "over-protection"
- Less control of navigation; for example, links might open new windows leading away from the original location
- Mental models and help systems might focus on understanding underlying concepts rather than narrow tasks
- Coding of color, typography, and sound maximize information to provide greater information density

15.3.2.5 Long- versus Short-Term Time Orientation

In the early 1980s, shortly after Hofstede first formulated his cultural dimensions, criticism convinced him that another dimension was needed to explain East Asian countries. After additional research with the Chinese Culture Connection (1987), he developed a new survey and extracted a fifth cultural dimension. LTO seemed to play an important role in Asian countries (influenced by Confucian philosophy over thousands of years) that shared these beliefs:

- A stable society requires unequal relations.
- Social life is governed by the *five relationships* (parent/child; elder sibling/younger sibling; husband/wife; friend/friend; and ruler/subjects).
- The extended family is the prototype of all social organizations; consequently, older people (parents) have more authority than younger (and men more than women).

- Virtuous behavior to others means not treating them as one would not like to be treated.
- Virtuous behavior in work means trying to acquire skills and education, working hard, and being frugal, patient, and persevering.

Western countries, by contrast, were more likely to promote equal relationships, emphasize IDV, focus on treating others as you would like to be treated, and find fulfillment through creativity and self-actualization. When Hofstede and Bond developed a survey specifically for Asia and reevaluated earlier data, they found that long-term orientation cancelled out some of the effects of masculinity/femininity and UA. They concluded that Asian countries are oriented to practice and the search for virtuous behavior while Western countries are oriented to belief and the search for truth. Of the 23 countries compared, the following showed the most extreme values (ranked in parentheses):

118	China (ranked 1)
80	Japan (4)
29	United States (17)
0	Pakistan (23)

High-LTO countries might emphasize the following aspects of UI design:

- Content focused on experience-based knowledge, practice, and practical value
- Relationships as the main source of information and credibility
- Patience in achieving results and goals

Low-LTO (short-term time orientation) countries might emphasize the contrary:

- Content focused on analytic knowledge, logical truth, and strong claims and assertions
- Rules and logic as the basis of information and credibility
- Desire for immediate results and achievement of goals

15.3.2.6 Cultural Dimensions and Design

Hofstede noted that some cultural relativism is necessary: it is difficult to establish absolute criteria for what is noble and what is disgusting. There is no escaping bias; all people develop cultural values based on their environment and early training as children. Not everyone in a society fits the cultural pattern precisely, but there is enough statistical regularity to identify trends and tendencies. These trends and tendencies should not be treated as defective or used to create negative stereotypes but recognized as different patterns of values and thought. In a multicultural world, it is necessary to cooperate to achieve practical goals without requiring everyone to think, act, and believe identically. In fact, contradiction should be viewed as a source of creativity.

Hofstede's cultural differences were applied to website design by Gould, Zakaria, and Yusof (2000) and Marcus and Gould (2000). Although appearances have changed in many of the sites, certain features related to metaphors, mental models, and navigation have survived for long periods of time. A study by Marcus (2005) showed that over a three-year period, websites associated with Sabena and British Airways changed dramatically but appeared to reflect different orientations to UA. The main air-booking areas had approximately the same number of links, but the UK site had twice as many links surrounding the central area, indicating that more complexity was tolerated.

Although this analysis has focused on Hofstede's dimensions, other authors have used other sets of dimensions (up to 762 in number) sometimes mixing several theorists. Marcus and Baumgartner (2004a and b) published a summary of Baumgartner's thesis, which surveyed approximately 60 experts worldwide on their views of the best dimensions from 11 theories for evaluating UIs. Twenty-nine dimensions emerged, from which five dimensions seemed to be most esteemed: (1) high versus low context, (2) technology attitudes and status, (3) UA, (4) time perception, and (5) authority perception. While the optimum set of dimensions continues to be debated, more publications and conference presentation within the UI community advocate the use of culture dimensions (or aligned concepts like cultural cognitive style) to analyze and design UIs.

In recent years, even the concept of usability has been the subject of cultural analysis. Frandsen-Thorlacius et al. (2009) showed that Danish and Chinese users differ substantially in their understanding of the concept of "usability." Chinese users include more association with "fun" and "appeal."

A growth area for research and analysis has been an interest in studying the differences of Web 2.0 applications, social networking, and mobile devices in regard to culture differences. User testing of mobile device usage, especially in daily use contexts is challenging. However, mobile-device eye-tracking equipment, such as that available from Mangold (see Figure 15.4) and other suppliers, and mobile-user message collection and management software, such as the Info-Pal tool from HP Labs (Jain 2010), promise to make the study and development of products/services more sophisticated and sensitive to local needs and cultural differences.

This review of cultural dimensions raises many issues about UI design, especially for the web and for mobile devices:

- How formal or rewarding should interaction be?
- What will motivate different people? Money? Fame? Honor? Achievement?
- How much conflict can people tolerate in content or style of argumentation?

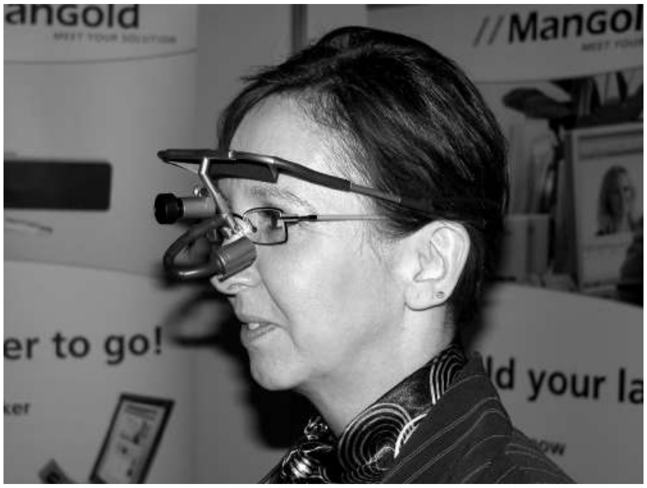


FIGURE 15.4 Example of a head-mounted display for conducting eye tracking of users during testing of mobile-phone applications. The unit is attached to a small portable device that the user carries. (Courtesy of Mangold International, Arnstorf, Germany).

- Should sincerity, harmony, or honesty be used to make appeals?
- What role exists for personal opinion versus group opinion?
- How well are ambiguity and UA received?
- Will shame or guilt constrain negative behavior on social media?
- What role do community values play in individualist versus collectivist cultures?
- Does the objective of distance learning change what can be learned in individualist versus collectivist cultures? Should these sites focus on tradition? Skills? Expertise? Earning power?
- Should online teachers or trainers act as friends, experts, or gurus?
- Would job sites differ for individualist versus collectivist cultures?
- Should there be different sites for men and women in different cultures?
- Would personal webcams be okay? Not okay?
- How much advertising hyperbole is likely to be tolerated?
- Would an emphasis on truth as opposed to practice and virtue require different types of social media for Western or Asian audiences?

To make cross-cultural theory an accepted element of UI design, we need to make it feasible to develop multiple culturally distinguished versions of interfaces and websites in a cost-effective manner, perhaps through templates or through specific versioning tools. As personal computer, web, and mobile products/services continue to develop globally, exploring and exploiting these dimensions of culture will become a necessity, not an option, for successful theory and practice.

15.4 CONCLUSIONS AND FUTURE RESEARCH ISSUES

To achieve culturally sensitive, successful global access to UIs provides many design challenges in the UI-development process. Progress in technology increases the number and kinds of functions, data, platforms, and users of computer-based communication media. The challenge of enabling more people and more kinds of people to use this content and these tools effectively will depend increasingly upon appropriately localized solutions. By recognizing the need for, and benefit to users of UI designs intended for international and intercultural markets, developers will achieve greater success and increased profitability through the global distribution and increased acceptance of their products and services.

The recommendations provided in this chapter include an initial set of heuristics that will assist developers in achieving global solutions to their product/service development. Design methodologies must support globalization throughout the development process. In addition, it is likely that some international and intercultural references will change rapidly, requiring frequent updating of designs. One urgent need is a collection of ROI case studies that show compelling business value of focusing on globalization/localization of UIs, especially in regard to cross-cultural issues. The subject-matter equivalent of an earlier study showing the ROI of usability in general (Marcus 2005) would benefit the field. Future work on global UI design may also address the following issues:

1. How many different kinds of user interfaces should be/can be prepared? Might global UIs be designed to account for different kinds of intelligence? Gardner (1985) identified the following dimensions of intelligence. These dimensions suggest users might have varying strengths of conceptual competence with regard to using UIs on an individual basis, but these might also vary internationally, or interculturally, due to influences of language, history, or other factors:

   - Verbal/image comprehension
   - Word/image fluency
   - Numerical/graphical fluency
   - Spatial visualization
   - Associative memory
   - Perceptual speed
   - Reasoning
   - Interpersonal awareness
   - Self-awareness

2. How might content management systems account for and adjust the metaphors, mental models, and navigation designed precisely for different cultures that might differ by such dimensions as age, gender, national or regional group, or profession? Further, what means can be developed to enable these variations to be produced in a cost-effective manner

using templates and a content-management system that can handle culture bases? Marcus has posed this issue earlier as a question to the UI analysis/design community (Marcus 1993a and b). The topic is discussed broadly in DelGaldo and Nielsen (1996).

The taxonomic analyses of global issues for UIs, the theoretical basis for their component selection, the criteria for their evaluation, and their design methodology have all emerged in the UI-development field. Articles about the impact of culture differences on UI design and techniques of ethnographic analysis appear ever more frequently in primary industry publications and in professional conferences such as those of ACM/SIGHCI, Human–Computer Interface International, UPA, American Anthropologists' Association, IWIPS, and others. The lively exchanges to be found on Anthropologists in Design's Internet discussions likewise attest to the growing numbers of professionals involved in this cross-disciplinary practice (see URL references for resources). Designers should be aware of the scope of the activity, know sources of insight, and incorporate professional techniques in their development process in order to improve the value and success of their international and intercultural computer-mediated products and services.

ACKNOWLEDGMENTS

The authors acknowledge Marcus et al. (1999) and Marcus and Gould (2000), on which this chapter was originally based. Mr. Marcus thanks his staff at Aaron Marcus and Associates, Inc., his coauthor Dr. Emile W. Gould, the editors of this book, and Dr. Constantine Stephanidis, the editor of *User Interfaces for All*, for their assistance in preparing this and earlier versions of this chapter. The authors also acknowledge DelGaldo and Nielsen (1996), Fernandes (1995), and Nielsen (1990), which provided a basis for some points raised in this chapter; and the advice of Peter Simlinger, Dipl.-Ing, Director, International Institute for Information Design, Vienna, Austria; Prof. Andreas Schneider, Information Design Department, Tama Art University, Tokyo, Japan; and Dr. Nuray Aykin, formerly New School, New York.

REFERENCES

Adler, N. J. 2002. *International Dimensions of Organizational Behavior.* 4th ed. New York: Southwestern.

American Institute of Graphic Arts (AIGA). 1981. *Symbol Signs*. New York: Visual Communication Books, Hastings House.

Associated Press. 2001. Saudi Arabia issues edict against Pokemon. *San Francisco Chronicle*, March 27, F2.

Aykin, N. 2000. (personal communication, 12 March 2000).

Aykin, N., ed. 2005. *Usability and Internationalization of Information Technology*. Mahwah, NJ: Lawrence Erlbaum.

Aykin, N., and A. E. Milewski. 2005. Practical issues and guidelines for international information display. In *Usability and Internationalization of Information Technology,* ed. N. Aykin, 21–50. Mahwah, NJ: Lawrence Erlbaum.

Barboza, D. 2006. The rise of Baidu (That's Chinese for Google). *The New York Times.* nytimes.com (accessed June 25, 2010).

BBC News. 2001. Saudi Arabia bans Pokemon. *BBC News.* news.bbc.co.uk/2/hi/middle_east/1243307.stm (accessed June 25, 2010).

Carroll, J. M. 1999. Using design rationale to manage culture-bound metaphors for international user interfaces. In *Proceedings of the International Workshop on Internationalization of Products and Services (IWIPS)*, 125–31. Rochester, NY.

Chavan, A. L. 1994. A Design Solution Project on Alternative Interface for MS Windows. Unpublished master's thesis. London, UK: Royal College of Ave.

Chavan, A. L. 2005. Another culture, another method. In *Proceedings of the 11th Human Computer Interaction International Conference*. Las Vegas, NV: CD-ROM.

China Hush. 2010. "Why Renren is better than Facebook", 5 April 2010. http://www.chinahush.com/2010/04/05/why-renren-is-better-than-facebook/ (accessed October 5, 2011).

Chinese Culture Connection. 1987. Chinese values and the search for culture-free dimensions of culture. *J Cross Cult Psychol* 18:143–64.

Chiu, L. H. 1972. A cross-cultural comparison of cognitive styles in Chinese and American children. *Int J Psychol* 7:235–42.

Choong, Y., and G. Salvendy. 1999. Implications for design of computer interfaces for Chinese users in mainland China. *Int J Hum Comput Interact* 11(1):29–46.

Clausen, C. 2000. *Faded Mosaic: The Emergence of Post-Cultural America*. Chicago, IL: Ivan R. Dee Publisher.

Death Penalty News. 2010. Saudi Arabia: Lebanese national sentence to death for sorcery. 18 March, 2010. http://deathpenaltynews.blogspot.com/2010/03/saudi-arabia-lebanese-national.html.

del Galdo, E. M., and J. Nielsen. 1996. *International User Interfaces*. New York: John Wiley.

Dong, Y. 2007. A cross-cultural comparative study on users' perception of the webpage: With the focus on cognitive style of chinese, korean and american. Masters Thesis. Korea Advanced Institute of Science and Technology, Daejeon, South Korea.

Fernandes, T. 1995. *Global Interface Design: A Guide to Designing International User Interfaces*. Boston, MA: AP Professional.

Ferraro, G. 2006. *The Cultural Dimension of International Business*. 5th ed. Upper Saddle River, NJ: Pearson Prentice Hall.

Forrester Research, Inc. 1998. *JIT web localization.* forrester.com (accessed July 4, 1998).

Forrester Research, Inc. 2001. *The Global User Experience*. Cambridge, MA: Forrester Research, Inc.

Forrester Research, Inc. 2009. *Translation and Localization of Retail Web SitesMaximizing the International Experience through Tailored Offerings.* forrester.com/rb/Research/translation_and_localization_of_retail_web_sites/q/id/54629/t/2 (accessed June 25, 2010).

Frandsen-Thorlacius, O., et al. 2009. Non-universal usability? A survey of how usability is understood by Chinese and Danish users. In *Proc. ACM Conf, SIG Human-Computer Interaction*, 41–58. Boston, MA.

Friedman, T. L. 2005. *The World is Flat: A Brief History of the Twenty-First Century.* New York: Farrar, Staus, and Giroux.

Gardner, H. 1985. *Frames of Mind, the Theory of Multiple Intelligences*. New York: Basic Books.

Gisèle Foucault. 2010. Personal interview with Emilie Gould, 12 March 2010.

Goode, E. 2000. How culture molds habits of thought. *New York Times*, August 8, D1ff.

Gould, E. W. 2001. More than content: Web graphics, crosscultural requirements, and a visual grammar. In *Proceedings of the Ninth International Conference on Human Computer Interaction 2001 (HCI 2001)*, Vol. 2, 506–9. New Orleans, LA.

Gould, E. W. 2005. Synthesizing the literature on cultural values. In *Usability and Internationalization of Information Technology*, ed. N. Aykin, 79–121. Mahwah, NJ: Lawrence Erlbaum.

Gould, E. W. 2007. "Only famous companies I would ever buy": Understanding how people learn to trust web sites. In *Proceedings of the 12th HCI International Conference.* Beijing, China. Heidelberg: Springer. [Electronic version]

Gould, E. W. 2009. Intercultural usability surveys: Do people always tell "the truth"? In *Proceedings of the 13th Human Computer Interaction International Conference*. San Diego, CA: CD-ROM.

Gould, E. W., N. Zakaria, and S. A. M. Yusof. 2000. Applying culture to website design: A comparison of Malaysian and US websites. In *Proceedings of 2000 Joint IEEE International and 18th Annual Conference on Computer Documentation (IPCC/SIGDOC 2000),* 161–71. New York: IEEE Press.

Hall, E. T. 1976. *Beyond Culture.* New York: Anchor Books.

Helft, M. and D. Barboza. 2010. Google shuts China site in dispute over censorship. *The New York Times*, 23 March 2010, A1.

Hendrix, A. 2001. The nuance of language. *San Francisco Chronicle*, April 15, A10.

Herskovitz, J. 2000. J-Pop takes off: Japanese music, movies, TV shows enthrall Asian nations. *San Francisco Chronicle*, December 26, C2.

Hoffman, P. 2010. UPA Professional Conference. Munich, Germany. May 24–28, 2010.

Hofstede, G. 1997. *Cultures and Organizations: Software of the Mind, Intercultural Cooperation and Its Importance for Survival*. New York: McGraw-Hill.

Honold, P. 1999. Learning how to use a cellular phone: Comparison between German and Chinese users. *J Soc Tech Commun* 46:196–205.

International Organization for Standardization. 1989. *Computer Display Color (Draft Standard Document 9241-8)*. Geneva, Switzerland: Author.

International Standards Organization. 1990. *ISO 7001: Public Information Symbols*. Geneva, Switzerland: The American National Standards Institute (ANSI).

International Standards Organization. 1998. ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs)—Part 11: Guidance on usability. iso.org/iso/iso_catalogue (accessed June 25, 2010).

ISO (International Standards Organization). 2010. ISO/IEC 8859-1:1998 Information technology—8-bit single-byte coded graphic character sets—Part 1: Latin alphabet No. 1. iso.org (accessed June 25, 2010).

Jain, J. 2010. InfoPal: A system for conducting and analyzing multimodal diary studies. In *Proceedings of the Usability Professionals Association, Annual Conference,* May 27, Munich, Germany, in preparation.

Kim, J. H., & K. P. Lee. 2007. Culturally adapted mobile phone interface design: Correlation between categorization style and menu structure. MobileHCI '07: Proceedings of the 9th international conference on Human computer interaction with mobile devices and services, 379–382. New York: ACM Press.

Kim, J., and J. Y. Moon. 1998. Designing towards emotional usability in customer interfaces: Trustworthiness of cyber-banking system interfaces. *Interact Comput* 10:1–29.

Knight, N., and R. E. Nisbitt. 2007. Culture, class, and cognition: Evidence from Italy. *J Cogn Cult* 7:283–91.

Kuhn, A. 2005. Mulling workdays, weekends in Iraq. npr.org/templates/story/story.php?storyId=4540715 (accessed June 25, 2010).

Kwintessential. 2010. Text expansion or contraction in translation. kwintessential.co.uk/translation/articles/expansion-retraction.html (accessed June 25, 2010).

Lee, K. P. 2010. Culture, interface design, and design methods for mobile devices. In Marcus, A., Roibás, A. C., & Sala, R. (Eds.), Mobile TV: Customizing content and experience, 37–66. London: Springer.

Letowt-Vorbeck, H. 2010. Mobile user experience: South Africa culture insights. In *Proceedings of the Usability Professionals Association, Annual Conference*, May 27, Munich, Germany, in preparation.

Marcus, A. 1995. Principles of effective visual communication for graphical user interface design. In *Readings in Human-Computer Interaction,* ed. R. Baecker, J. Grudin, W. Buxton, and S. Greenberg, 2nd ed., 425–41. Palo Alto, CA: Morgan Kaufman.

Marcus, A. 1998. Metaphor design in user interfaces. *J Comput Doc* 22:43–57.

Marcus, A. 2003. 12 myths of mobile UI design. *Software Development Magazine*, May, 38–40.

Marcus, A. 2005. User interface design's return on investment: Examples and statistics. In *Cost-Justifying Usability*, Chapter 2 in ed. R. G. Bias and D. J. Mayhew, 2nd ed., 17–39. San Francisco, CA: Elsevier.

Marcus, A. 2006. Cross-cultural user-experience design. In *Diagrammatic Representation and Inference, Proc., 4th International Conference, Diagrams 2006, Stanford, CA*, ed. D. Barker-Plummer, R. Cox, and N. Swoboda, 16–24. Berlin, Germany: Springer-Verlag.

Marcus, A., and V. J. Baumgartner. 2004a. Mapping user interface design components vs. culture dimensions in corporate websites. *Visible Lang* 38(1):1–65.

Marcus, A., and V. J. Baumgartner. 2004b. A practical set of culture dimension for evaluating user interface designs. In *Proceedings of the Sixth Asia-Pacific Conference on Computer-Human Interaction (APCHI 2004),* 252–61. Rotorua, New Zealand.

Marcus, A., and E. W. Gould. 2000. Crosscurrents: Cultural dimensions and global web user-interface design. *Interactions* 7(4):32–46.

Masuda, T., and R. E. Nisbett. 2001. Attending holistically vs. analytically: Comparing the context sensitivity of Japanese and Americans. *Journal of Personality and Social Psychology* 81:922–34.

McSweeney, B. 2002. Hofstede's model of national cultural differences and their consequences: A triumph of faith-a failure of analysis. *Human Relations* 55(1):89–118.

Nicol, G. 1994. *The Multilingual World Wide Web.* xml.coverpages.org/nicol-multiwww.html (accessed June 25, 2010).

Nielsen, J., ed. 1990. Designing user interfaces for international use: Vol. 13. In *Advances in Human Factors/Ergonomics.* Amsterdam, the Netherlands: Elsevier Science.

Nisbett, R. E. 2003. *The Geography of thought: How Asians and Westerners Think Differently … and Why*. New York: Free Press.

Nisbett, R. E., and A. Norenzayan. 2002. Culture and cognition. In *Stevens' Handbook of Experimental Psychology, Third Edition, Volume Two: Memory and Cognitive Processes*, ed. D. Medin and H. Pashler. New York: John Wiley & Sons.

Olgyay, N. 1995. *Safety Symbols Art*. New York: Van Nostrand Reinhold.

Ona, Y., and B. Spindle. 2000. Japan's long decline makes one thing rise: Individualism. *Wall Street Journal*, December 30, A1.

Pierce, T. 1996. *The International Pictograms Standard.* Cincinnati, OH: S. T. Publications.

Prensky, M. 2001a. Digital natives, digital immigrants. *Horizon* 9(5): 1–6. www.marcprensky.com/writing (accessed February 1, 2010).

Prensky, M. 2001b. Digital natives, digital immigrants; *Part II:* Do they really *think* differently? *Horizon* 9(6):1–9. www.marcprensky.com/writing (accessed February 1, 2010).

Raby, M. 2007. Microsoft begins compliance with EU regulations. *TG Daily*. tgdaily.com/business-and-law-features/34471-microsoft-begins-compliance-with-eu-regulations (accessed June 25, 2010).

Samovar, L. A., R. E. Porter, and E. R. McDaniel, eds. 2008. *Intercultural Communication: A Reader.* 12th ed. New York: Wadsworth.

San Francisco Chronicle. 2001. Saudi Arabia issues edict against Pokemon. March 27, F2.

Schwartz, S. H. 1994. Beyond individualism/collectivism: New cultural dimensions of values. In *Individualism and Collectivism: Theory, Method, and Applications,* ed. U. Kim, H. C. Triandis, C. Kagitcibasi, S.-C. Choi, and G. Yoon, 85–119. Thousand Oaks, CA: Sage.

Snopes.com. 2007. Come alive! snopes.com/business/misxlate/ancestor.asp (accessed June 25, 2010).

Stephanidis, C., ed. 2000. *User Interfaces for All: Concepts, Methods, and Tools.* Boca Raton, FL: CRC Press.

Stephanidis, C., ed. 2009. *The Universal Access Handbook (Human Factors and Ergonomics)*. Boca Raton, FL: CRC Press.

Stille, A. 2002. *The Future of the Past*. New York: Farrar, Straus, and Giroux.

Suchman, L. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication.* Cambridge: Cambridge University Press.

Trompenaars, F., and C. Hampden-Turner. 1998. *Riding the Waves of Culture: Understanding Diversity in Global Business*. New York: McGraw-Hill.

Unicode. 2010. CLDR—Unicode Common Locale Data Repository. cldr.unicode.org/ (accessed June 25, 2010).

Victor, D. A. 1992. *International Business Communication*. New York: HarperCollins.

Warschauer, M. 2003. *Technology and Social Inclusion; Rethinking the Digital Divide.* Cambridge, MA: MIT Press.

Why Renren is better than Facebook. 2010. *China Hush.* chinahush. com (accessed June 25, 2010).

Yardley, J. 2000. Faded mosaic nixes idea of "cultures." *U.S. San Francisco Examiner*, August 7, B3.

You, I. K. 2009. *Cognitive Style and Its Effects on Generative Session Comparing Korean and European Participants*. Unpublished master's thesis. Daejeon, KR: KAIST (Korea Advanced Institute of Science and Technology).

URLS AND OTHER INFORMATION RESOURCES

(Latest version available by e-mail to Marcus at aaron.marcus@amanda.com)

ACM/SIGCHI Intercultural Issues database: www.acm.org/sigchi/intercultural/

ACM/SIGCHI Intercultural listserve: chi-intercultural@acm.org

American National Standards Institute (ANSI): www.ansi.org

Anthropologists in Design: http://groups.yahoo.com/group/anthrodesign/

Bibliography of Intercultural publications: www.HCIBib.org//SIGCHI/Intercultural

China National Standards: China Commission for Conformity of Elect. Equip. (CCEE) Secretariat; 2 Shoudu Tiyuguan, NanLu, 100044, P. R. China; Tel: 186-1-8320088, ext. 2659, Fax: 186-1-832-0825.

Cultural comparisons: www.culturebank.com, www.webofculture.com, www.iir-ny.com

Digital divide: www.digitaldivide.gov, www.digitaldivide.org, www.digitaldividenetwork.org

Globalization and Internet language statistics: language: www.euromktg.com/globstats/, www.sapient.com, www.worldready.com/biblio.htm

Glossary, six languages: www.bowneglobal.com/bowne.asp? page59&language51

International Standards Organization (ISO): http://www.iso.ch/

Internationalization providers: www.basistech.com, www.cij.com, www.Logisoft.com

Internationalization resources: www.world-ready.com/r_intl.htm, www.worldready.com/biblio.htm

Internet users survey, Nua: www.nua.ie/surveys/how_many_online

Japan Info. Processing Society; Kikai Shinko Bldg., No. 3-5-8 Shiba-Koen, Minato-ku, Tokyo 105, Japan; Tel: 181-3-3431-2808, Fax: 181-3-3431-6493.

Japanese Industrial Standards Committee (JISC); Min. of Internat. Trade and Industry; 1-3-1, Kasumigaseki, Chiyoda-ku, Tokyo 100, Japan; Tel: 181-3-3501-9295/6, Fax: 181-3-3580-1418.

Java Internationalization: http://java.sun.com/docs/books/tutori

Johnston, K., and P. Johal. 1999. The Internet as a 'virtual cultural region': Are external cultural classification schemes appropriate? *Internet Research* 9(3):178–86.

Localization Industry Standards Organization (LISA): www.lisa.org

Localization providers: www.Alpnet.com, www.Berlitz.com, www.globalsight.com, www.lhsl.com, www.Lionbridge.com, www.Logisoft.com, www.Logos-usa.com, www.translations.com, www.Uniscape.com

Machine translation providers: www.babelfish.altavista.com, www.IDC.com, www.e-Lingo.com, Lernout & Hauspie <www.lhsl.com>, www.Systransoft.com

Microsoft global development: www.eu.microsoft.com/globaldev/

Simplified English: userlab.com/SE.html

Unicode: www.unicode.org/, www–4ibm.com/software/developer/library/glossaries/unicode.html

World-Wide Web Consortium: www.w3.org/International, www.w3.org/WAI

RECOMMENDED READINGS

Alvarez, G. M., L. R. Kasday, and S. Todd. 1998. How we made the website international and accessible: A case study. In *Proceedings of the 4th Human Factors and the Web Conference*. Holmdel, NJ: CD-ROM.

Batchelor, D. 2000. *Chromophobia*. London: Reaktion Books.

Boxer, S. 2001. Vivid color in a world of black and white. *New York Times*, April 28, A15ff.

Brain, D. 2005. Syllabus for course in sociology of culture at New College, FL. http://www.ncf.edu/brain/courses/culture/culture_syl05.htm (accessed December 31, 2005).

Coriolis Group. 1998. *How to Build a Successful International Website*. Scottsdale, AZ: Author.

Cox Jr., T. 1994. *Cultural Diversity in Organizations*. San Francisco: Berrett-Koehler.

Crystal, D. 1987. *The Cambridge Encyclopedia of Language*. Cambridge, UK: Cambridge University Press.

Daniels, P. T., and W. Bright, eds. 2000. *The World's Writing Systems*. Sandpoint, ID: MultiLingual Computing.

Day, D. L. 2000. Gauging the extent of internationalization activities. In *Proceedings of the 2nd International Workshop on Internationalization of Products and Systems*, 124–36. Baltimore, MD: Backhouse Press.

Day, D. L., E. M. del Galdo, and G. V. Prabhu, eds. 2000. Designing for global markets 2. In *Proceedings of the 2nd InternationalWorkshop on Internationalisation of Products and Systems*. Rochester, NY: Backhouse Press.

Day, D. L., and L. M. Dunckley. 2001. *Proceedings of the Third International Workshop on Internationalisation of Products and Systems (IWIPS 2001)*. Milton Keynes, UK.

Day, D. L., V. Eves, and E. del Galdo. 2005. Designing for global markets 7: Bridging cultural differences. In *Proceedings of the International Workshop on Internationalization of Products and Services 2005*. Amsterdam: Grafisch Centrum Amsterdam.

Doi, T. 1973. *The Anatomy of Dependence*. New York: Kodansha-International.

Doi, T. 1986. *The Anatomy of Self: The Individual Versus Society*. New York: Kodansha International.

Dreyfuss, H. 1966. *Symbol Sourcebook*. New York: Van Nostrand Rhinehold.

Earley, P., C. Soon, and A. Soon. 2003. *Cultural Intelligence: Individual Interactions Across Cultures*. Stanford, CA: Stanford University Press.

Elashmawi, F., and P. R. Harris. 1998. *Multicultural Management 2000: Essential Cultural Insights for Global Business Success*. Houston, TX: Gulf.

Evers, V., et al., eds. 2003. *Proceedings of the Fifth International Workshop on Internationalisation of Products and Systems (IWIPS 2003)*. Berlin, Germany: Products and Systems Internationalization.

Fetterman, D. M. 1998. *Ethnography: Step by Step*. 2nd ed. Thousand Oaks, CA: Sage.

Forrester Research, Inc. 1998. JIT Web localization. www.Forrester.com (accessed July 4, 1998).

French, T., and A. Smith. 2000. Semiotically enhanced Web interfaces: Can semiotics help meet the challenge of cross-cultural design? In *Proceedings of the 2nd International Workshop on Internationalization of Products and Systems*, 23–38. Baltimore, MD. Rochester, NY: Backhouse Press.

Graham, T. 2000. *Unicode: A Primer*. Sandpoint, ID: MultiLingual Computing and Technology.

Gudykunst, W. B. 2003. *Cross-Cultural and Intercultural Communication*. Thousand Oaks, CA: Sage.

Gudykunst, W. B. 2005. *Theorizing About Intercultural Communication*. Thousand Oaks, CA: Sage Publications.

Hall, E. T. 1969. *The Hidden Dimension*. New York: Doubleday.

Harel, D., and P. Girish. 1999. Global User Experience (GLUE), Design for cultural diversity: Japan, China, and India. Designing for Global Markets. In *Proceedings of the First International Workshop on Internationalization of Products and Systems (IWIPS-99)*, 205–16. Rochester, NY: Backhouse Press.

Harris, J., and R. McCormack. 2000. *Translation is Not Enough*. San Francisco, CA: Sapient.

Harris, P. R., and R. T. Moran. 1993. *Managing Cultural Differences*. Houston, TX: Gulf.

Hofmann, P. 2010. *Making Icons Make Sense: Solving Symbols for Global Audiences*. Presentation, Usability Professionals Association, Munich, Germany.

Hoft, N. L. 1995. *International Technical Communication: How to Export Information About High Technology*. New York: John Wiley and Sons, Inc.

Inglehart, R. F., M. Basanez, and A. Moreno, eds. 1998. *Human Values and Beliefs: A Cross-Cultural Sourcebook*. Ann Arbor, MI: University of Michigan Press.

International Standards Organization. 1993. *ISO 7001: Public Information Symbols: Amendment 1*. Geneva, Switzerland: The American National Standards Institute (ANSI).

Kimura, D. 1992. Sex differences in the brain. *Sci Am* 267(3):118–25.

Kohls, L. R., and J. M. Knight. 1994. *Developing Intercultural Awareness: A Cross-Cultural Training Handbook*. 2nd ed. Yarmouth: ME: Intercultural Press.

Konkka, K., and A. Koppinen. 2000. Mobile devices: Exploring cultural differences in separating professional and personal time. In *Proceedings of the Second International Workshop on Internationalization of Products and Systems*, 89–104. Rochester, NY: Backhouse Press.

Kuhn, A. 2005. Mulling weekends, workdays in Iraq. National Public Radio Web Archive. http://www.npr.org/templates/story/story.php?storyId54540715 (accessed January 12, 2006).

Kurosu, M. 1997. Dilemma of usability engineering. In *Design of Computing Systems: Social and Ergonomics Considerations, Volume 2. Procedings of the 7th International Conference on Human-Computer Interaction HCI International '97*, ed. G. Salvendy, M. Smith, and R. Koubek, 555–8. Amsterdam: Elsevier.

Lee, Y. S., Y. S. Ryu, T. L. Smith-Jackson, D. J. Shin, M. A. Nussbaum, and K. Tomioka. 2005. Usability testing with cultural groups in developing a cell phone navigation system. In *Proceedings of the 11th International Conference on Human-Computer Interaction (HCII '05)*, Las Vegas, NV: CD-ROM.

Leventhal, L. 1996. Assessing user interfaces for diverse user groups: Evaluation strategies and defining characteristics. *Behav Inf Technol* 15(3):127–38.

Lewis, R. 1991. *When Cultures Collide*. London: Nicholas Brealey.

Lingo Systems. 1999. The guide to translation and localization Los Alamitos, CA IEEE Computer Society, ISBN 0-7695-0022-6.

LISA. 1999. The Localization Industry Primer. The Localization Industry Standards Association, (LISA) 7, rute du Monastère, 1173 Féchy, Switz., 35. www.lisa.org.

Marcus, A. 1992. *Graphic Design for Electronic Documents and User Interfaces*. Reading, MA: Addison-Wesley.

Marcus, A. 1993a. Designing for diversity. In *Proceedings of the 37th Human Factors and Ergonomics Society*, 258–61. Seattle, WA.

Marcus, A. 1993b. Human communication issues in advanced UIs. *Commun ACM* 36(4):101–9.

Marcus, A. 2000. International and intercultural user interfaces. In *User Interfaces for All*, ed. C. Stephanidis, 47–63. New York: Lawrence Erlbaum Associates.

Marcus, A. 2001. User interface design for air-travel booking: A case study of Sabre. *Inf Des J* 10(2):186–206.

Marcus, A. 2003. 12 myths of mobile UI design. *Software Development Magazine*, May, 38–40.

Marcus, A. 2005a. User interface design and culture. In *Usability and Internationalization of Information Technology*, ed. N. Aykin, 51–78. New York: Lawrence Erlbaum.

Marcus, A., and E. W. Gould. 2000. Crosscurrents: Cultural dimensions and global web user interface design. *Interactions* 7:32–46.

Marcus, A., Gould, and E. Chen. 1999. Globalization of user interface design for the Web. In *Proceedings of the 5th Human Factors and the Web Conference*. Gathersburg, MD: CD-ROM.

Matsumoto, D., and J. LeRoux. 2003. Measuring the psychological engine of intercultural adjustment: The intercultural adjustment potential scale (ICAPS). *J Intercult Commun* 6:27–52.

Matsumoto, D., M. D. Weissmann, K. Preston, B. R. Brown, and C. Kupperburd. 1997. Context-specific measurement of individualism-collectivism on the individual level: The individualism-collectivism interpersonal assessment inventory. *J Cross Cult Psychol* 6(28):743–67.

Murphy, R. 2005. Getting to know you. Fortune Small Business. http://www.fortune.com/fortune/smallbusiness/technology/articles/0,15114,1062892-1,00.html (accessed November 6, 2005).

Neustupny, J. V. 1987. Communicating with the Japanese. *The Japan Times*. Tokyo, Japan.

Nisbett, R. E., P. Kaipeng, C. Incheol, and A. Norenzayan. 2001. Culture and systems of thought: Holistic versus analytical cognition. *Psychol Rev* 108:291–310.

Ota, Y. 1973. *Locos: Lovers Communications System (in Japanese)*. Tokyo, Japan: Pictorial Institute.

Ota, Y. 1987. *Pictogram Design*. Tokyo, Japan: Kashiwashobo Publishers.

Peng, K. 2000. *Readings in Cultural Psychology: Theoretical, Methodological and Empirical Developments During the Past Decade (1989–1999)*. New York: John Wiley.

Perlman, G. 1999. ACM SIGCHI intercultural issues. In *Proceedings of the Second International Workshop on Internationalization of Products and Systems*, 183–95. Rochester, NY: Backhouse Press.

Pierce, T. 1996. *The International Pictograms Standard*. Cincinnati, OH: S. T. Publications.

Prabhu, G. V., B. Chen, W. Bubie, and C. Koch. 1997. Internationalization and localization for cultural diversity. In *Design of Computing Systems: Cognitive Considerations. Vol. 1: Proceedings of the 7th International Conference on Human-Computer Interaction (HCI International '97)*, ed. G. Salvendy, 149–52. Amsterdam: Elsevier.

Prabhu, G. V., and E. M. delGaldo, eds. 1999. Designing for global markets 1. In *Proceedings of the First International Workshop on Internationalization of Products and Systems*, 226. Rochester, NY: Backhouse Press.

Prabhu, G. V, and D. Harel. 1999. GUI design preference validation for Japan and China: A case for KANSEI engineering? In *Proceedings of the 8th International Conference on Human-Computer Interaction (HCI International '99)*. Munich, Germany.

Schwartz, S. H. 2004. Mapping and interpreting cultural differences around the world. In *Comparing Cultures, Dimensions of Culture in a Comparative Perspective*, ed. H. Vinken, J. Soeters, and P. Ester, 43–73. Leiden, the Netherlands: Brill.

Shahar, L., and D. Kurz. 1995. *Border Crossings: American Interactions with Israelis*. Yarmouth, Maine: Intercultural Press.

Singh, N. 2004. From cultural models to cultural categories: A framework for cultural analysis. *J Am Acad Bus* 5(1/2):1–8. [Vol. 5, Nos. 1 & 2].

Singh, N., and D. W. Baack. 2004. website adaptation: A cross-cultural comparison of U. S. and Mexican websites. *J Comput Mediat Commun* 9(4), http://JCMC.Indiana.edu/vol9/issue4/singh_back.html (accessed September 16, 2008).

Singh, N., and H. Matsuo. 2002. Measuring cultural adaptation on the Web: A content analytic study of U.S. and Japanese websites. *J Bus Res* 57(8):864–72.

Spradley, J., and D. McCurdy. 1998. *The Cultural Experience: Ethnography in Complex Society*. Long Grove, IL: Waveland Press.

Stille, A. 2001. An old key to why countries get rich: it's the culture that matters, some argue anew. *New York Times*, January13, 81.

Storti, C. 1994. *Cross-Cultural Dialogues: 74 Brief Encounters with Cultural Difference*. Yarmouth, ME: Intercultural Press.

Tannen, D. 1990. *You Just Don't Understand: Women and Men in Conversation*. New York: William Morrow and Company, Inc.

Thomas, D., and K. Inkson. 2004. *Cultural Intelligence: People Skills for Global Business*. San Francisco: Berrett-Koehler Publishers.

Traugott, M. 2008. Syllabus for course in contemporary sociological theory. University of California at Santa Cruz. http://ic.ucsc.edu/~traugott/socy105a/syllabus.html (accessed October 5, 2011).

Vickers, B. 2000. Firms push to get multilingual on the Web. *Wall St Journal*, November 22, B11A.

Würtz, E. 2005. A cross-cultural analysis of websites from high-context cultures and low-context cultures. *J Comput Mediat Commun* 11(1):article 13.

Yeo, A. W. 2001. Global-software development lifecycle: An exploratory study. In *Proceedings Computer-Human Interaction Conference 2001*, 104–11. Seattle, WA.

16 Speech and Language Interfaces, Applications, and Technologies

Clare-Marie Karat, Jennifer Lai, Osamuyimen Stewart, and Nicole Yankelovich

CONTENTS

16.1 INTRODUCTION

A spoken interface for a computer often emulates human–human interaction by calling on our inherent ability as humans to speak and listen. While human speech is a skill we acquire early and practice frequently, getting computers to map sounds to actions and to respond appropriately with either synthesized or recorded speech is a massive programming undertaking. Because we all speak a little differently from each other, and because the accuracy of the recognition is dependent on an audio signal that can be distorted by many factors, speech technology, like the other recognition technologies, lacks 100% accuracy. When designing a spoken interface, one must design to the strengths and weaknesses of the technology to optimize the overall user experience.

The goal for a spoken user interface is to emulate a human dialog convincingly enough that the person interacting with the computer can use what he has learned in a lifetime of conversations. Successful communication is when the sender and receiver of the message achieve a shared understanding. Human-to-human conversations are characterized by turn-taking, shifts in initiative, as well as verbal and nonverbal feedback to indicate understanding. Herb Clark (1993) says, "Speaking and listening are two parts of a collective activity." Because language use is deeply ingrained in human behavior, successful speech interfaces should be based on an understanding of the different ways that people use language to communicate. Speech applications should adopt language conventions that help people know what they should say next and avoid conversational patterns that violate standards of polite, cooperative behavior.

There are excellent examples of speech user interfaces that emulate effective conversational partners. In these systems, the computer "speaker" appears to remember contextual information and gives the impression of understanding what the user is saying. For example, several airlines use speech applications to handle telephone reservations or lost baggage tracking (Cohen, Giangola, and Balogh 2004). These systems are a substantial improvement over earlier interactive voice response (IVR) systems that relied solely on telephone keypad input. Instead of pressing 1 for this and 2 for that, users can speak natural language phrases such as "I'd like to travel from Boston to New York."

A crucial factor in determining the success of a spoken application is whether there is a clear benefit to using speech technology. Speech is best used when it enables something such as conducting transactions (e.g., checking bank balances) that cannot otherwise be done over the telephone when a computer keyboard is not available, or for providing real-time automated translation services in a business setting for a user in conversation with another person speaking a different language. In general, it is effective to use speech applications for situations when speech can enable a task to be done more efficiently, for example, when a user's hands and eyes are busy doing another task. Likewise, speech input is useful when there is no keyboard available for text entry, or if people have a physical disability that limits use of their hands, or if they are just not comfortable typing. Speech output is particularly liberating for people with visual impairments. In addition, it provides a way of communicating information if the user is in a divided attention state such as driving, and it can be used to grab users' attention or to embody a particular personality for a computer system or character.

Although speech seems like it might be the ideal way to communicate with a computer anytime, there are situations when it is best not to use a speech user interface. For example, speech output is ineffective for delivering large amounts of information. Not only is it difficult for users to maintain the information in short-term memory, but people can read much faster than they can listen. Speech input can also be problematic when the speaker is not in a private environment or when there are other voices in the background that might interfere with the speech recognition.

The success of a spoken interaction with a computer depends not only on the task and the motivation of the user, but it is also dependent on the physical devices being used. Speech input and output capabilities of devices vary a lot. For example, personal computers (PCs) provide good-quality audio subsystems and speakers, and there are a variety of microphones that can be used with them that perform very well under quiet conditions. The audio channels for other devices are not on par with PCs. While most smartphones offer speech input capabilities, the quality of their microphones and audio subsystems degrade the speech signal, resulting in poor recognition performance for many applications. Also handheld devices have insufficient computing resources to enable large vocabulary speech recognition processing. For telephony applications, when the input device is either a cell phone or a land line, the speech recognition engines are usually deployed on large servers. The accuracy of telephony systems for a given task is normally lower than that for a similar PC configuration with a headset microphone in a quiet environment. Background noise, signal degradation, poor cellular connection, and the application of compression techniques can substantially reduce the performance of speech recognition systems. Nevertheless, careful design can compensate for many of these problems and lead to successful interaction with a telephone-based speech application. In this chapter, we will also introduce and discuss the current status of language translation applications. When considering speech-to-speech or text-to-text translation applications, it is even more important to address the issues raised above in the design process as there is the additional complexity of working with two or more languages and the need for graceful error recovery within the application.

Speech recognition technology can be used for the following types of tasks:

- *Composition.* Composition tasks have the creation of a document such as word processing documents, e-mails, or instant messaging text as their primary goal. Composition includes dictating the text and fixing any recognition errors.
- *Transcription.* Transcription is similar to composition in that a document is created from speech, but it differs by virtue of the fact that the primary user task is parallel to the creation of a document. Broadcast news, business meetings, and calls are examples of situations in which having a permanent, textual record of the speech is valuable. A digital (textual) record of the conversation is searchable, readable by the deaf and supports advanced business intelligence applications such as data mining. These are all examples of transcription tasks.
- *Transaction.* The third type of interaction, and a major focus of this chapter, is one in which users have as their goal the completion of one or more transactions, rather than the creation of a document

or a permanent record of a conversation. Examples of transactional applications include financial account management such as trading stocks, e-commerce applications such as the purchase of computer equipment, searching for information on the Internet, or controlling the environment.

- *Translation*. The fourth type of interaction, an emerging and critical domain in the interconnected world in which we live, involves the use of computers to translate written or spoken communications from one language to another. The process whereby a computer (e.g., a smartphone or handheld device) systematically transfers or translates the meaning of a text string or a speech utterance from one language to another is called machine translation. The translation task can range from the translation of written documents (newspapers, articles, transcriptions, etc.) to the asynchronous or nearly real-time communication between two people facilitated through computer technology. In one of the most technically challenging scenarios of use, machine translation enables Speaker A's communication in language A to be translated to Speaker B's language B so that the two speakers can read or hear the communication of the other person in his or her desired language, understand the communication, and respond appropriately in an ongoing dialog.
- *Collaboration*. The final type of conversational task is collaboration. Collaborative conversational applications are characterized by tasks that result in human-to-human communication. An example of this is the use of speech recognition as an input modality to an instant messaging application. Human-to-human collaboration is also one of the major applications of machine translation.

As can be seen from the types of tasks that speech technology can be applied to, speech applications may involve the use of speech technology appropriate to one or more of the tasks described above. The set of examples and scenarios described above illustrate the building block nature of the design solutions for speech technology. For example, a speech application may require a design solution for users that handles both composition and collaboration. A second more challenging speech and language application may handle several tasks to complete the user's goal (e.g., working with service providers from other countries to create travel plans and pay for a trip): composition with translation, transaction with real-time translation, and collaboration with real-time translation. A usable design solution may require the combined use of different speech technologies to address the user and system requirements, and the output from one task in an application may be the input for the next step in the application solution.

Designing a speech user interface is similar to designing any other interface for human–computer interaction. A good design relies on applying principles of user-centered design, and many of these same principles and techniques can be used with speech interfaces. In this chapter, we discuss the lifecycle of a speech application from the starting point of understanding how the current and emerging speech and language translation technologies work and their capabilities and limitations, and we discuss the human–computer interaction (HCI) process of crafting a spoken interaction, including understanding the user requirements for speech, knowing which technology to use, selecting a dialog style, and designing the prompts. All these steps are completed in an iterative testing process with target users to refine the speech or language application.

We have expanded the scope of the chapter in this edition by introducing the emerging domain of language translation. Language translation applications build on the technological methods used in speech applications. These methods can be viewed as building blocks in the speech and language technologies domain. Language and translation applications break new ground with informational and collaboration applications. HCI methods and processes are used in the design of language and translation applications and examples of the design process for these new types of applications are provided. The additional challenges of translation interfaces as compared with speech interfaces are discussed and research to address some of these challenges is reviewed. We conclude the chapter with a look to the future of research and development in speech and language technologies and applications.

16.2 SPOKEN INTERFACE LIFECYCLE

Once a designer has clearly established that including speech technology for input, output, or both, is an appropriate design decision, he/she must begin to define how speech best fits into the accomplishment of the task. In order to do this, the designer must first have a clear understanding of the task. With the task well understood, the designer needs to overlay speech onto that model by listening to how people speak in the domain. Observing humans interacting and speaking to accomplish the task may result in additional refinements to the model. The next step is to select the appropriate speech technology, or combination of technologies, for the task, context of use, and user group. The heart of the work for a speech interface designer is to craft the prompts that the system will "speak" and to prepare the system to recognize what the users will most likely say in response. Lastly, there should be a series of tests for the system, initially in the lab and ultimately out in the real world with users accomplishing real tasks. In between the various forms of testing, a designer needs to allocate time to refine the design and make changes based on what he learns from the results of the testing. In this chapter, we will discuss certain steps in greater detail than others. These steps include selecting the technology, and the design of the system including the user prompts.

Thus, the first step in the lifecycle of a spoken interface is modeling the task the users will be accomplishing. This requires the designer to sketch out the logical steps that make up the task, along with the pieces of information that need to be exchanged in order to complete each step. With a speech application, each interaction between the system and a user is often referred to as a "turn." Thus the *task model* tries to capture the expected number of turns as well as the vocabulary that is required to support the information exchange during those turns. It is important for the designer to consider alternate flows because not all users will necessarily approach the task in the same way. For example, when making a flight reservation, some users will select a flight based exclusively on time schedules, whereas others will opt for airline loyalty and price points, accepting any flight time that meets their criteria for loyalty and cost.

The model needs to operate within the set of constraints for the application. Constraints usually include the business goals for creating the speech application, as well as the requirements resulting from the environment that users may find themselves in (e.g., noisy environment) and the users themselves (e.g., the majority of users will be more than 65 years of age and uncomfortable with technology). For example, if a business goal is to cross-sell related items when a customer makes a purchase by calling into a speech-enabled call center application, the designer will want to incorporate a turn that involves the system suggesting related items that are on sale today.

16.3 UNDERSTANDING SPEECH AND LANGUAGE TECHNOLOGIES

Although good interaction design in a speech application can compensate for some short-comings in speech technology, if a certain baseline level of accuracy is not achieved, the application will probably not succeed. Accuracy depends on the choice of the underlying speech technology, and making the best match between the technology, the task, the users, and the context of use. Automatic speech recognition (ASR) can have explicitly defined rule-based grammars or use statistical grammars such as a language model. Usually a transactional system uses explicitly defined grammars while dictation systems or natural language understanding (NLU) systems use statistical models. The designer will also have to decide whether to use synthetic speech in the system or not. Synthetic speech, also known as text-to-speech (TTS), is speech produced by a computer. Given that today's synthesizers still do not sound entirely natural, the choice whether to use synthesized output, recorded output, or no speech output can be a difficult one. The next section discusses the various technologies in more detail.

16.3.1 How Does Automatic Speech Recognition Work?

ASR systems work by analyzing the acoustic signal received through a microphone connected to the computer (see Figure 16.1). The user speaks some text and the microphone captures the acoustic signal as digital data, which is then analyzed by an acoustic model and a language model. The different speech recognition systems on the market differ in the building blocks (e.g., phonemes) that they use to analyze the acoustic signal data. The analysis uses algorithms based on Hidden Markov Models, a type of algorithm that uses stochastic modeling to decode a sequence of symbols, to complete the computations (Rabiner 1989; Roe and Wilpon 1993). After the acoustic analysis is complete, the system analyzes the resulting strings of building block data using a language model that contains a base vocabulary and any specific domain topics (e.g., computer, medical) that may have been added. When the analysis is completed, the text appears on the computer screen.

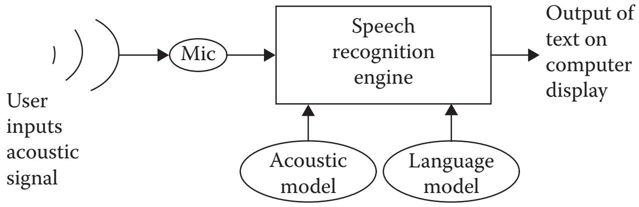

User inputs acoustic signal
Mic
Speech recognition engine
Output of text on computer display
Acoustic model
Language model

FIGURE 16.1 Overview of human–computer interaction model for speech recognition.

Speech recognition systems require computers with an approximate minimum of 200-MHz processor, 32 MB of RAM, 300 MB of available hard disk, and a 16-bit sound card with a microphone input jack and good recording. These requirements enable local decoding of the recognized speech. It is possible to have a system, such as a small pervasive device, capture a user's speech and then decode it on a remote server (e.g., Price and Sears 2005) and return the decoded text to the user, albeit with a short time delay. Multiple users can work with one installation of a speech recognition system. In some systems, a user creates a personal voice model and logs on with their individual user name, and the system uses their personalized speech files for the recognition processing. Each user can also create several different user voice models in order to achieve the best recognition rates in environments with different levels of background noise (e.g., home, office, and mobile work locations). For results on the effectiveness of creating and using different user voice models, see Price et al. (2006).

16.3.2 Current Capabilities and Limitations of Automatic Speech Recognition Systems

Karat et al. (1999) report ASR system accuracy rates in the mid-90s for "in vocabulary" words (i.e., words in the 20,000 word vocabulary included with the software) in people's initial use of continuous speech recognition software. With more frequent use, error rates of 2%–5% are common. Karat et al. (1999) tested three commercially available speech recognition systems in 1998 with users in initial and extended use and found initial use data of 13.6 corrected words per minute with an average of one incorrect, missing, or extra word for every 20 words and one formatting error every 75 words. Improved ASR system accuracy and higher user productivity were measured when users used more recent versions of the speech recognition software (Sears et al. 2000, Feng 2006) because users are able to correct errors more easily and quickly. The product also provided a better quality microphone to reduce the number of recognition errors that occurred in the first place.

In general, there are three types of errors in ASR systems (Halverson et al. 1999; Karat et al. 2000). Users can make direct errors where they mis-speak, stutter, or press the wrong key. Second, users can make errors of intent, where they decide to restate something. The third type of error is an indirect error, where the speech recognition system misrecognizes what the user says. The indirect errors are difficult to detect during proofreading. All three types of errors can lead to cascading errors where in the process of correcting one error, others occur. These types of errors challenge the usability of ASR speech systems for users and their adoption and continued use by users. An HCI case study of speech research and design considerations for usability in ASR speech systems is provided in Karat and Karat (2010). Motivated users (users with learning disabilities, disabled users) can certainly learn to use ASR systems productively over time. Some improvements have been made in the gracefulness of error recovery for users. However, these systems do not yet display the usability required for widespread adoption by the population.

Most telephony systems are speaker-independent (i.e., no personalized training of the voice models required) speech recognition systems. They are also usually server based and must handle the signal degradation that occurs across the telephone lines. Telephony systems can be created from a combination of speech recognition and natural language processing (NLP) technologies. In telephony systems, a dialog manager component works with the speech recognition software to handle the course of the conversation with the user. The system provides feedback to the user through the dialog manager using recordings of either human voice or TTS. Telephony systems have capabilities such as "barge in" and "talk ahead" that enable the user to redirect the action of the system and complete multiple requests before being prompted for additional information necessary to complete the task. Conversational telephony systems include IVR systems and new systems built using voice XML (please see Section 16.3.5 on NLP for description of the technology). These systems work well and allow users to efficiently complete desired tasks.

16.3.3 How Does Text-to-Speech Work?

TTS synthesis enables computers or other electronic systems such as telephones to output simulated human speech. Synthetic speech is based on the fields of text analysis, phonetics, phonology, syntax, acoustic phonetics, and signal processing. There is a hierarchy of the quality and effectiveness of speech synthesis. The base level of achievement is to produce speech synthesis that is intelligible by human beings. The second level is to produce speech synthesis that simulates the natural qualities of human speech. The third level of speech synthesis is to produce synthesized speech that is personalized to the person it is representing; that is, it has the intonation of the particular person's speech being represented. The fourth and highest level of achievement in synthesized speech is to produce speech based on a person's own voice recordings so that the speech sounds just like the actual person being represented. Currently, speech synthesis technology has achieved the base level of quality and effectiveness, and concatenated synthesis can simulate the natural quality of human speech, although at great expense. The third and fourth levels of speech synthesis technology are the focus of research in laboratories around the world.

There are two types of speech synthesis commercially available today, concatenated synthesis and formant synthesis, which is the most prevalent type. Concatenated synthesis uses computers to assemble recorded voice sounds into speech output. It sounds fairly natural but can be prohibitively expensive for many applications, as it requires large disk storage space for the units of recorded speech and significant computational power to assemble the speech units on demand. Concatenated synthesizers rely on databases of diphones and demisyllables to create the natural sounding synthesized speech. Diphones are the transitions between phonemes. Demisyllables are the half-syllables recorded from the beginning of a sound to the center point, or from the center point to the end of a sound (Weinschenk and Barker 2000). After the voice units are recorded, the database of units is coded for changes in frequency, pitch, and prosody (intonation and duration). The coding process enables the database of voice units to be as efficient as possible.

Formant synthesis is a rule-based process that creates machine-generated speech (see Figure 16.2). A set of phonological rules is applied to an audio waveform that simulates human speech. Formant synthesis involves two complex steps. The first includes the conversion of the input text into a phonetic representation. The second encompasses the production of sound based on that phonetic representation. In the first step, the text is input from a database or file and is normalized so that any symbols or abbreviations are resolved as full alphabetic words. To convert the words into phonemes, a pronunciation dictionary is used for most words and a set

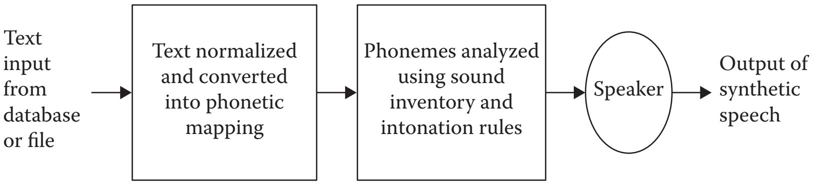

Text input from database or file
Text normalized and converted into phonetic mapping
Phonemes analyzed using sound inventory and intonation rules
Speaker
Output of synthetic speech

FIGURE 16.2 Model of text-to-speech synthesis.

of letter-to-sound rules is used for word exceptions not available in the dictionary. In the second step, the phonemes are analyzed using a sound inventory and intonation rules about pitch and duration. The speech synthesis is the resulting output that is heard by users through a speaker or headphone.

The quality of the synthetic speech should be evaluated along the lines of its acceptability, naturalness, and intelligibility. It is important to ask users to evaluate different speech against each other because these qualities are always relative (Francis and Nusbaum 1999). Although these subjective differences in opinion will probably always exist, a study by Lai, Wood, and Considine (2000) showed that there were no significant differences in comprehension levels for longer messages (i.e., with a word length ranging from 100 to 500 words) among five major commercial TTS engines.

16.3.4 Current Capabilities and Limitations of Speech Synthesis Software

Formant synthesis produces speech that is highly intelligible but sounds unnatural. However, it has the power to produce nearly unlimited speech inexpensively from a resource point of view. The limitation of using formant synthesis is the complexity of the required linguistic rules to produce accurate speech output. Utilizing domain-specific information and assumptions produces a substantial improvement in the synthesizer's prosody. Prosody refers to speech elements such as intonation, duration, pitch, rate, loudness, or rhythm. Users will be able to comprehend the speech at a higher rate and will perceive the voice to be more natural.

Some applications of concatenated synthesis attempt to reduce costs by basing the voice recordings on whole words. However, these systems often sound unnatural and unevenly paced, which makes the synthetic speech hard to understand or remember. An application of concatenated synthesis should be done correctly or not at all. Also it is advisable when using synthetic speech to not mix it with human speech as this tends to degrade comprehensibility (Gong and Lai 2003).

In the last several years, numerous business organizations have successfully deployed speech systems that are a mixture of IVR and speech recognition technologies to handle customer service calls. For example, these speech systems are successfully used to handle prescription refills for healthcare organizations, track a package or lost luggage, as well as enable customers to check on the status of their financial accounts in banking/finance organizations. These systems are now complemented by websites so customers have a choice in the type of contact channel to use in interactions with the organizations.

16.3.5 How do Natural Language Processing and Natural Language Understanding Work?

NLP refers to a wide range of processing techniques aimed at extracting, representing, responding to and ultimately understanding the semantics of text. NLU is an area of NLP focused on the understanding of natural language text. It is the process of analyzing text and taking some action based on the meaning of the text. We include any technology that allows a user to communicate with a system using a language that is not rigidly structured (i.e., a "formal" language). The focus of this chapter is on systems where communication between the user and the system has constructs similar in grammar and dialog to the language of everyday human–human communication.

As a technology, NLU is independent from speech recognition, although the combination of the two yields a powerful HCI paradigm. When combined with NLU, speech recognition transcribes an acoustic signal into text, which then is interpreted by an understanding component to extract meaning. In a conversational system, a dialog manager will then determine the appropriate response to give the user. Communication with the user can take place through a variety of modalities including speech input and output, text input and output, handwriting, or some combination of these modalities. Figure 16.3 shows a block diagram of a prototypical multimodal conversational system that allows speech and keyboard natural language input, and speech and graphical user interface (GUI) text output.

NLU has been an active area of research for many decades. The promise of NLU lies in the "naturalness" of the interaction. Because humans have deep expertise in interacting with each other through the use of language, it has been an implicit and explicit hypothesis in a wide array of research studies and technology-development efforts that leveraging a user's ability to interact using language will result in systems with greater usability. Designing systems that use natural interaction techniques mean the user is freed from learning the formal language of a system. Thus, instead of using formalisms (e.g., UNIX commands), scripting languages or graphical menus and buttons, users can engage in a dialog with the system. Less user training, more rapid development of expertise, and better error recovery are all promising aspects of systems that use NLU.

Dialog can be described as a series of related conversational interactions. It adds the richness of context and the knowledge of multiple interactions over time to the user interface, transforming natural language interfaces into conversational interfaces. Such interactions require the system to maintain a history of the interaction as well as the state of the interaction at all times (Chai 2001). Two important components, which must be represented in any dialog system, are user goals and the current context of the interaction. The history of the interaction must be evaluated against the user goal, with prompts to the user designed to acquire the necessary information required to satisfy a goal. Dialog technology can be embedded in an application to enable either user-initiated, system-initiated, or mixed-initiative conversations (see descriptions of these different dialog styles in Section 16.4.1). The conversations are guided by dialog management technology.

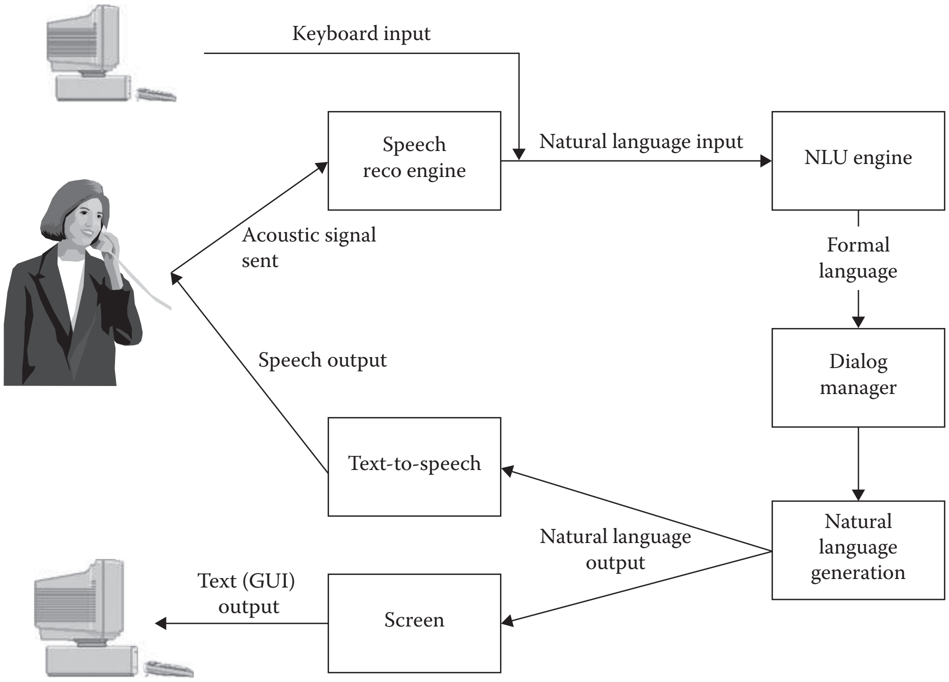


FIGURE 16.3 Diagram of a prototypical multimodal conversational system.

16.3.6 Current Capabilities and Limitations of Natural Language Processing and Natural Language Understanding Systems

Designing speech and language systems poses substantial challenges. Understanding these challenges and assessing the various trade-offs that must be made during the design process will help to produce the most effective interface.

By its nature, speech is *transient*. Once you hear it or say it, it is gone. By contrast, graphics are *persistent*. A graphical interface typically stays on the screen until the user performs some action. Listening to speech taxes users' short-term memory. Because speech is transient, users can remember only a limited number of items in a list and they may forget important information provided at the beginning of a long sentence. Users' limited ability to remember transient information has substantial implications for the speech interface design. In general, transience means that speech is not a good medium for delivering large amounts of information. The transient nature of speech can also provide benefits. Because people can look and listen at the same time, speech is ideal for grabbing attention or for providing an alternate mechanism for feedback. Imagine receiving a notification about the arrival of an e-mail message while working on a spreadsheet. Speech might give the user the opportunity to ask for the sender or the subject of the message. The information can be delivered without forcing the user to switch contexts.

Speech is also *invisible*. The lack of visibility makes it challenging to communicate the functional boundaries of an application to the user (Yankelovich 1996). In a graphical application, menus and other screen elements make most or all of the functionality of an application visible to a user. In contrast, in a speech application, it is much more difficult to indicate to the user what actions they may perform, and what words and phrases they must say to perform those actions.

Capturing and representing knowledge of a domain is a complex and labor-intensive process. As the scope of the application domain increases, it becomes increasingly difficult to build and maintain NLU applications. As a result, all successful examples of rich NLU interfaces have relatively narrow application domains. This has direct impact on the definition of user profiles for any given application. Simply stated, the narrower the conceptual, functional, syntactic, and lexical domains of the target user population, the greater the chance of building an NLU application that satisfies its users. There is currently no way to quantify and measure these characteristics, and determining whether a particular application represents a tractable problem in NLU is largely an issue of experience and instinct on the part of designers and engineers. An NLU engine along with speech input and output technologies are the building blocks for a conversational application. Now let's turn our attention to the emerging domain of language translation technologies.

16.3.7 How Does Machine Translation Work?

Language remains one of the major barriers in a linguistically diverse and globally connected world with an ever-increasing need for people to communicate and collaborate with each other. Thus, it is common to find, in formal settings like the United Nations, representatives of the various nations wearing head-phones through which they receive translations by a human translator or interpreter who works behind the scenes. Moreover, with recent advances in machine translation technology, computers (translation software programs on a PC, smartphones, or other handheld devices) are now also used to facilitate communication between people who speak different languages or to empower people to consume information that is typically only available in a language different from theirs. The process whereby a computer (e.g., smartphones or handheld devices) systematically transfers (translates) the meaning of a text string or a speech utterance from one natural language to another is called machine translation.

The success of machine translation is predicated on the computer system's ability to know all the relevant words or sentences along with their meaning in a source language which are then mapped to the corresponding words, sentences, and their meanings in a target language. This kind of mapping and its effectiveness (accuracy) remains an enormous challenge for machine translation systems and underscores the usability limitations of the technology. As a result, machine translation operates based on the notion of "domain." A domain refers to the thematic content that is required for the interaction in a specified topic or area of conversation. This includes Information Technology (IT Support for technical documents), product manuals, government documents, patents, travel or tourism, weather information, news items, and so on. Indeed, language translation is a complex natural linguistic activity due to the vast amount of variations in how we speak and interpret what is spoken. This makes domains very critical to the success of machine translation technology by defining a priori the domain of thematic content required in the translation corpus. As Goshawke, Kelly, and Wigg (1987) point out "we cannot ask for translations of poetry or subtle literary references or puns or jokes," because of the linguistic and cognitive features necessary for interpreting language including culture, context, and so on.

In general, machine translation domains are comparable with conversational tasks discussed in the speech technology section; however, there is one major difference: translation is a derivative in the conversational task or indirect (embedded) step in the communication process (see Table 16.1). Essentially, a user must first perform a task in the source language (native language), which is then automatically rendered in the target language through machine translation. Thus, the successful outcome of the translation into the target language is dependent on the accuracy or proper characterization from the source language. In light of this interaction flow, it will be beneficial to evaluate how conversational tasks apply (or are realized) through machine translation:

In light of these tasks or functions, it is important to note that millions of people use machine translation every day, mostly on the Internet, to view information in a foreign language. When it does not work, people often wonder how difficult machine translation can really be. In simple terms, the translation process consists of two parts:

1. Decoding the meaning of the text or speech in the source language
2. Re-encoding this meaning in the target language

To perform step 2, machine translation must know and be able to analyze all the features of the string being translated. This requires deep linguistic and cognitive knowledge of the morphology, semantics, syntax, idioms, and so on of the source language, as well as the culture and stylistics of its speakers. Similar in-depth knowledge is required for re-encoding the meaning in the target language. For machine translation technology to be successful, the requirement is for the computer to be able to mediate (translate) between two linguistic universes of the source and target languages. There are two major approaches to handling this requirement: rule based and statistical.

16.3.7.1 Rule-Based Machine Translation

A rule-based machine translation (RBMT) combines dictionary entries of the words in a language plus the "linguistic rules" governing the use of those words which are then encoded into a computer to operate over input data (TAUS Report 2007). Linguistic rules refer to information pertaining

TABLE 16.1
Types of Conversational Tasks

Conversational Task	Objective	Machine Translation
Composition	Create a document (e.g., Word, e-mail, instant messaging)	Document creation: an entire document or a text already created in a source language is translated into a target language
Transcription	The rendering of the content of a document, done in parallel to the composition or creation to create a permanent textual record (e.g., broadcast news, business meetings, calls)	Informational function: translation of content either in real time (e.g., broadcast news, viewing foreign websites), or through batch processing (e.g., viewing meetings, calls, etc.)
Transaction	Completion of one or more transactions	Transactional function: real-time translation to facilitate the completion of one or more transactions between two people who speak different languages (e.g., in tourism, for booking a room in Beijing during the 2008 Olympics)
Collaboration	Aid human–human communication	Collaboration function: real-time translation to facilitate collaboration between two people who speak different languages (e.g., the use of web-based multilingual translator for instant text or speech messaging)

to morphology (word structure such as tense inflection, singular versus plural inflection, etc.), semantics (word meanings as encoded through dictionary definitions), syntax (grammatical outline of how words are combined to form correct phrases and sentences), homonyms (lexical information about word ambiguity that may be triggered depending on various contexts), and so on. All these rules are itemized as formal features of a linguistic system, and then packaged in software programs for translation.

16.3.7.2 Statistical Machine Translation

A statistical machine translation (SMT) works with textual data, not predefined language rules. It processes text by means of pattern-matching algorithms that do not contain any formal "language rules" just a collection of patterns or words that make up the bilingual text corpora to which the statistical methods apply (TAUS Report 2007). Essentially, the system looks at and stores all the linear patterns of words (groups of two, three, or more words) in a text in one language. It then tries to "match" a correlating pattern in a translated version of this same text. This matching can be exact (where the patterns are exactly the same) or fuzzy (where the patterns do not match 100%). In principle, a SMT "learns" from a body of existing translations in order to identify plausible patterns of language in both texts, without reference to any linguistic rules. In this regard, one crucial component necessary for teaching the SMT to recognize or "learn" recurring patterns is the "translation memory." A translation memory is the repository of all the exact matches that exist in parallel text corpora. Quite often, the SMT system relies very heavily on the knowledge-bases provided by the translation memory, and these patterns therein can be used to translate segments of new texts, which will often contain similar groups of words (TAUS Report 2007).

There are some major differences between these two approaches. RBMT requires intensive human-effort (skilled linguists) to come up with the extensive dictionary of words along with the associated large sets of linguistic rules spanning several levels: phonological, morphological, semantic, and syntactic. Most machine translation systems (and especially commercial deployments) now use SMT, whose major drawback is the difficulty in getting enough parallel corpora (of the right kind) to support the learning approach. However, both approaches are similar based on the fact that in order to be successful, all machine translation systems need to be customized. According to the TAUS Report (2007), customization in RBMT involves the adaptation of the dictionary (or glossary) of terms to include new insights or new user requirements, and the linguistic rules. In SMTs, this usually involves retraining the system on a translation memory corpus plus relevant dictionary terms (or glossary). Most practitioners and researchers focus on SMT and so we will only focus on SMT in the subsequent discussion of machine translation technology.

There are two general kinds of SMT systems: speech-to-speech and text-to-text, with the obvious difference attributable to the nature of the input/output mode, whether speech or text.

16.3.7.2.1 Speech-to-Speech

In a speech-to-speech system, two people who speak different languages can engage in real-time communication over the telephone or any socket connection over the Internet (see Figure 16.4). For example, Speaker A speaks in English and the audio (speech) of the utterance is recognized by the ASR (comprising the acoustic and language model) for English, which converts the speech to text. If the speech-to-speech system is a conversational one, then the text string is passed on to the NLU component which assigns or extracts the meaning, otherwise an NLU component is not necessary. Subsequently, the NLU output (form and meaning) is sent to the machine translation component for translation into

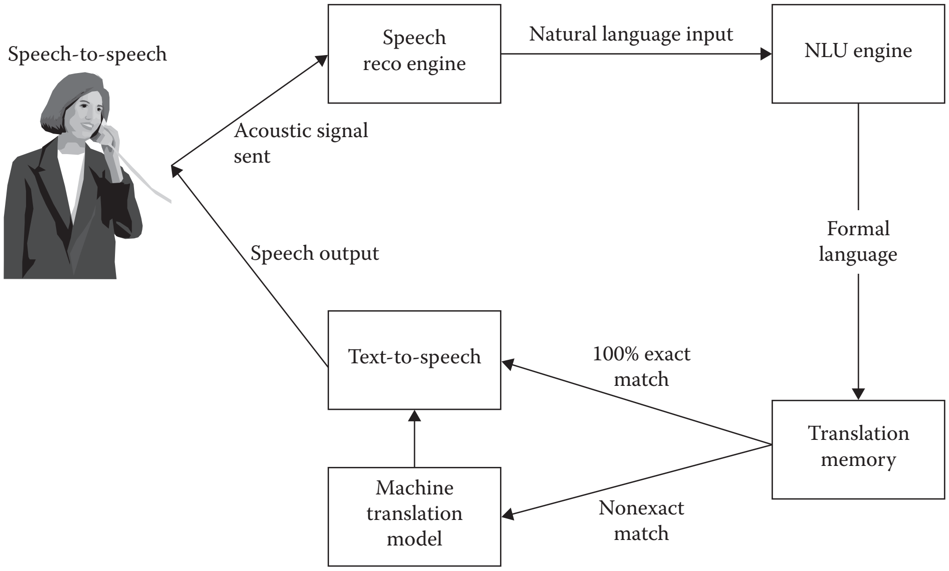


FIGURE 16.4 Diagram of a prototypical speech-to-speech machine translation system.

a target language, for example, Chinese. In its most rudimentary form, there are two subsystems inside the machine translation component: translation memory and machine translation model. The incoming text string initially goes to the translation memory to see if there is an exact match. If one exists, then that is offered as the translation which is presented to the Speaker B. However, if no exact match exists (and the business logic rules out accepting fuzzy matches), then the text string is passed on to the machine translation model where the statistical methods apply to generate the "best guess" translation. Subsequently, the translated utterance (in Chinese) is presented to Speaker B through the Chinese TTS. The entire process happens in real time with latency around 3 seconds or less on the average between the dialog turns. Stewart et al. (2009a) have shown that one problem arising from this multistage process of speech recognition, followed by translation, and then voice synthesis, is that it strips away many of the features of the variations in voice dynamics. More specifically, the eventual TTS output comes out mostly flat without any systematic variations in the tone of voice appropriate to the content of the translation, culture, or language. However, most users are currently preoccupied with the accuracy of the translation and will readily accept poor TTS quality.

16.3.7.2.2 Text-to-Text

A text-to-text system allows people who speak different languages to view information or communicate using various web-based clients or widgets on the Internet. Text-to-text systems work pretty much like the speech-to-speech with the difference being that the input mode is only text (see Figure 16.5). This somewhat simplifies the way text-to-text systems work because there is no need for an acoustic model in the ASR, only the language model is required and the recognized text string is passed on the machine translation component. Also, most text-to-text systems do not include an NLU component (nonconversational systems). With these exclusions in place, then, in general, the same translation process with speech-to-speech also applies: First it looks for exact match in the translation memory and then defaults to the machine translation engine. Thereafter, as there is no need for TTS, the translated text string is passed on the Speaker B in the user interface on the client or widget. With the simplification, the entire process happens in real time with little or no latency (in milliseconds) between the dialog turns.

16.3.8 Current Capabilities and Limitations of Machine Translation Systems

The Internet has truly made the world a global village because it has been able to remove or mitigate the traditional space and time constraints from collaboration. However, along with this growth and adoption is the increasing demand to consume content or collaborate in local languages as only 8%–10% of the world's population of 6.1 billion speaks English (Aykin 2005). Furthermore, there is also the trend that more people travel to different parts of the world (for business or vacation) where English is not the first language. Finally, as globally integrated (multinational) companies, like IBM, with linguistically diverse teams and customers, expand their business to various parts of the world, there is the need to make their services available in the local language and to also empower their employees to engage in cross-national, cross-lingual collaboration. Within the enterprise, there is usually a deluge of online content that grows faster than with which the finite set of professional human translators can ever cope. In addition, companies are faced with the ever-increasing cost of translating online content and continue to seek ways to mitigate the cost of human translation. The translation problem is further compounded because, oftentimes, work in process content becomes obsolete even before the translation cycle is complete, and this content is published by the human professional translators. These conditions (and more) have greatly influenced the development of machine translation solutions based on SMT algorithms.

There is currently no real-world deployment of telephone or Internet-based (non face-to-face) speech-to-speech translation systems. The problem may be attributed to many complex technical and usability issues including disambiguation, accuracy, and latency that get in the way of successful conversational interaction between two people who may be on different continents. Moreover, besides the research-turned commercial version of IBM's multilingual automatic speech-to-speech translator system (Gao et al. 2002, Zhou, Dechelotte, and Gao 2004), there are also very few successful implementations of face-to-face speech-to-speech translation

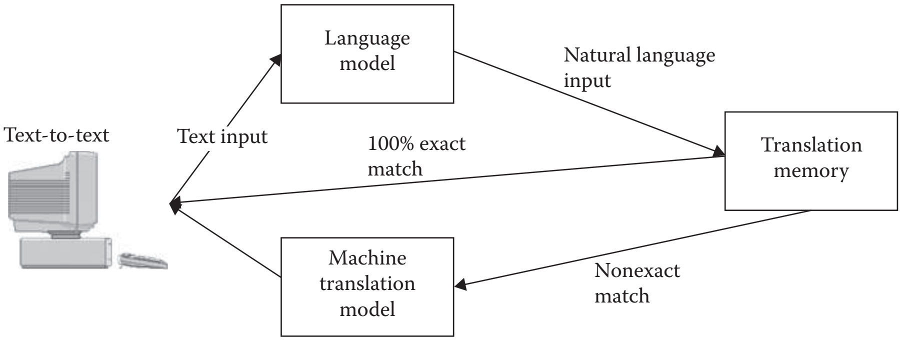


FIGURE 16.5 Diagram of a prototypical text-to-text machine translation system.

systems on devices like smartphones. Therefore, the promise of constructing robust speech-to-speech translations systems to empower people who speak different languages (and may physically be in different locations) to engage in cross-lingual communication is still a dream of speech and natural language research.

In contrast, there are very many successful implementations of text-to-text translation systems, especially those that are offered as widgets on web pages on the Internet and various online software programs (e.g., www.asiaonline.net, www.google.com/translate) for translating documents on the Internet. In addition, there are many commercial translation companies called Language Service Providers, for example, Language Weaver, Lionbridge, and so on, who also use text-to-text machine translation for translation localization, that is, for augmenting the human translation process where machine translation is used to do an initial pass of a document and then a human translator applies postediting rules for completing the document translation. There are also a few machine translation applications now available on smartphones for the business traveler that translate, for example, the most frequent questions or phrases that travelers use in designated locations. Against this background, there are several issues facing machine translation systems, in addition to the technical and usability limitations. The biggest one has to do with public perception and trust. For example, do they really work? This further drives the rapid adoption of this technology to domains where the cost of errors is low, that is, an appreciation for the ability of the technology to basically make information available that is not possible otherwise, like viewing IT or technical support information, news, weather, sports, but not legal or medical information (or any formal domain) where there is a higher demand for accuracy.

With an understanding of how speech and translation technologies work as well as their current capabilities and limitations, let's now consider the design of speech and language applications for users.

16.4 CRAFTING A SPOKEN INTERACTION

The first step in designing a speech interface is to educate yourself about the ways people speak in the domain of the task. The best approach is to find users who are doing an activity that is as close as possible to the target task. Keep in mind that the vocabulary and sentence structure used in graphical computer applications or in printed material may be quite different from the way people actually speak about a task. For example, imagine that you want to provide a speech interface to a calendar. Printed calendars and online calendars typically show day, week, and month views of calendar appointments. They always show days of the week and numbered dates. As soon as you listen to two people talk about appointments and scheduling, however, you learn that they only use numbered dates occasionally. The more common way to discuss dates with other people is to use relative descriptions: "What do you have a week from Monday? Are you busy this Thursday? What about the next day?" The concept of relative dates is completely absent from paper-based and online calendars. But when speaking, users will expect to be able to use them.

With a little creativity, you can find or create situations in which two people can talk about the target task. Although you might want to plan out the interactions a little bit, care must be taken not to put words in the mouths of the users. The idea is to allow people's natural social instincts to drive the dialog. If at all possible, record the natural dialogs. They will serve as the basis for the speech interface design. Listening in this way is called a *natural dialog study* (Yankelovich 2008). Using natural dialog studies, you can learn about concepts used when talking about the task that are absent in other media, common vocabulary used when talking about the task, tone of voice that is considered polite in the context of the task, sentence structure typical for the domain patterns of interaction, and methods people use to give one another feedback.

16.4.1 Dialog Styles

Once you have an understanding of your task and the way people speak in this domain, you will need to decide if your application would benefit most from a directed, user-initiated, or mixed-initiative dialog style.

16.4.1.1 Directed Dialog (System-Initiated)

This is the most commonly used style of interaction in speech-based telephony systems on the market today. With a directed dialog, the user is instructed or "directed" what to say at each prompt (Kamm 1994). Systems that use this style can often be recognized by the use of the word "say" in the prompts. "Welcome to ABC Bank. You can check an account balance, transfer funds, or pay a bill. Say Balance, Transfer, or Pay." The reason most systems use this dialog style is to increase the accuracy of the speech recognition. By telling the user what to say, the chances of the user speaking an out-of-vocabulary utterance are much lower. This style is sometimes referred to as system-initiated because the user plays a more passive role, waiting for the system to prompt him for what is needed.

16.4.1.2 User-Initiated

Unlike a directed dialog, which works well for novice users, user-initiated systems are intended for use by expert users. With this type of interaction style, the system is passive and the user is not prompted for specific information. For example,

Computer: How can I help you?
User: I would like to buy 100 shares of Amazon.com at the market price with funds from my current account number 345198.

These types of dialogs do not work well with new users who do not know what to say, and tend to have much lower recognition accuracy rates.

16.4.1.3 Mixed Initiative

The type of interaction that is easiest for users and feels the most natural is a mixed-initiative system. In this style of interaction, the system and the user take turns initiating the communication depending on the flow of the conversation and the status of the task. With a mixed-initiative system, the prompts can fluctuate between open-ended prompts and more directed prompts depending on the context. The following is a sample dialog:

Computer: What would you like to do? (*open ended*)
User: What is the price of Fidelity Magellan? (*user-initiated*)
Computer: Price of Fidelity Magellan is $36.05 per share. What else?
User: I want to buy 500 shares. (*which shares to buy is implied in the previous interaction*)
Computer: Confirming purchase of 500 shares of Fidelity Magellan fund, please say yes or no. (*directed*)
User: No, skip the purchase. I'd like to transfer $1,000 from my index fund (*user-initiated*).
Computer: Purchase canceled. To which fund would you like to transfer? *(prompt for missing information*)

In this example, we see the system functioning as a cooperative partner in the conversation.

16.4.2 Prompt Design

The challenges of working with speech can often be mitigated by well-designed prompts. Many factors must be considered when designing prompts, but the most important is assessing the trade-off between flexibility and performance. The more you constrain what the user can say to an application, the less likely they are to encounter recognition errors. However, allowing users to speak information flexibly can often speed the interaction (if recognition succeeds), feel more natural, and avoid forcing users to memorize commands. The following are some tips for creating useful prompts.

- Use *explicit directed prompts* when the user input must be tightly constrained. For example, after recording a message, the prompt might be "Say cancel, send, or review." This sort of prompt directs the user to say just one of those three keywords. Even mixed-initiative systems should fall back on this type of prompt when it is critical that the input be correctly recognized, as with transactions that involve transferring money. "Are you sure you want to transfer $1,000 from your savings to your checking account. Please say Yes or No."
- Use *implicit prompts* when the application is able to accept more flexible input. These prompts rely on conversational conventions to constrain the user input. For example, if the user says "Send mail to Bill," and "Bill" is ambiguous, the system prompt might be "Did you mean Bill Smith or Bill Jones?" Users are likely to respond with input such as "Smith" or "I meant Bill Jones." While possible, conversational convention makes it less likely that they would say "Bill Jones is the one I want."
- Using *variable prompts* is a good way to try to simulate a human–human conversation. Given a certain condition or state of the system (e.g., the ready state, or a system response to silence), it is preferable not to play the exact same system prompt every time. Subtle variations in the wording impart a much more natural feel to the interaction. Note the following possibilities for the ready state in an NLU system: "What now," "I'm ready to help," "What's next?"
- Another interaction that we can model on human speech is the use of *tapered prompts*. Tapering can be accomplished in one of two ways. If an application is presenting a set of data such as current quotes for a stock portfolio, drop out unnecessary words once a pattern is established. For example: "As of 15 minutes ago, Acme Industries was trading at 45 up ½, BioStartup was at 83 up ½, and TechGiant was at 106 down ¼." Tapering can also happen over time. That is, if you need to tell the user the same information more than once, make it shorter each time the function is used.
- One way to speed interaction for expert users and provide help for less-experienced users is to use *incremental prompts*. This technique involves starting with a short prompt. If the user does not respond within a time-out period, the application prompts again with more detailed instructions. For example, the initial prompt might be "Which service?" If the user says nothing, then the prompt could be expanded to "Say banking, address book, or yellow pages." Incremental prompts have been used successfully in a number of systems, but suffer from several problems. A first-time user is just as likely to say something to the first prompt, as they are to say nothing. This often results in a recognition error. Another common pitfall with incremental prompts is that they tend to cause collision errors where both the system and the user speak at the same time. It is not uncommon for timing problems to happen repeatedly once started, increasing the user's frustration along the way.

16.4.3 Providing Help

There are two basic types of help. In the first case, the user initiates by requesting help. Common queries for help in speech systems are "Help," "What can I say?" or "What are my choices?" It is always a good idea to support several variations of this request in the speech recognition grammar or language model. In the second type of help, the system detects that the user is in trouble and presents help automatically. Although not all forms of user problems are detectable by the system, one can presume if a series of errors or long silences occur that the user probably needs some assistance.

Help, when given, needs to be specific to the user's current task. There are few things as frustrating as being stuck and asking for help only to find that the information presented, while long and detailed, has no bearing on the current situation. A good approach for help is to provide the user with an example of a successful utterance just as he starts the downward spiral of repeated errors. If several errors have occurred in a row, the system should present the user with sample phrasing that would be expected to work, based on the context that he or she is in. For example, "You could say: I want to sell 300 shares of Ultrasoft." If you cannot determine the exact context, another option is to list the functions that are available at the current branch and have the user point the system in the right direction. Here is an example:

Computer: Let's start over. You can forward, reply, delete, or create a new message. What do you want to do?
User: I want to forward a message.
Computer: To forward a message, say the first and last name of another subscriber. For example, say "Forward this message to John Smith."
User: Forward this message to Alishia Grant.

If the system returns a recognition error now, it is easy to establish that the name is at fault because we already know the function. This dialog might continue as follows:

Computer: Oh dear. I didn't catch that name. Just say the first name.
User: Alishia.
Computer: I think you said Alice, is that correct?
User: No, that is not correct.
Computer: Thanks for your patience with this. Please spell the first name.
User: A L I S H I A

Now the system can repeat a similar process for the last name, or can look up all subscribers with a first name of Alishia and offer the user a set of possible last names to choose from.

16.4.4 Handling Errors

How a system handles recognition errors can dramatically affect the quality of a user's experience. If either the application or the user detects an error, an effective speech user interface should provide one or more mechanisms for correcting the error. Although this seems obvious, correcting a speech input error is not always easy! If the user speaks a word or phrase again, the same error could reoccur depending on the cause of the error (Yankelovich, Levow and Marx 1995).

Recognition errors can be divided into three categories: rejection, substitution, and insertion (Schmandt 1994, Ballentine and Morgan 1999). A rejection error occurs when the recognizer has no hypothesis about what the user said. A substitution error involves the recognizer mistaking the user's utterance for a different valid utterance, as when "send a message" is interpreted as "seventh message." With an insertion error, the recognizer either interprets noise as a valid utterance, or decodes multiple words when only one was spoken. This can be caused by other people talking nearby or by the user inadvertently tapping the telephone or microphone.

16.4.4.1 Rejection Errors

In handling rejection errors, you want to avoid the "brick wall" effect, when every rejection is met with the same "I didn't understand" response. Users get frustrated very quickly when faced with repetitive error messages. Instead, give *progressive assistance*: a short error message the first couple of times, and if errors persist, offer more detailed assistance. For example, here is one progression of error messages that a user might encounter: "Sorry?", "What did you say?", "Sorry. Please rephrase.", "I didn't understand. Speak clearly, but do not overemphasize.", "Still no luck. Wait for the prompt tone before speaking." Progressive assistance does more than bring the error to the user's attention; the user is guided toward speaking a valid utterance with successively more informative error messages that consider possible causes of the error.

16.4.4.2 Substitution Errors

Although rejection errors are frustrating, substitution errors can be damaging. If the user asks a weather application for "Kuai," but the recognizer hears "Good-bye" and hangs up, the interaction could be completely terminated. In situations like this, the system should explicitly verify that the user's utterance was correctly understood. Verification should be commensurate with the cost of the action that would be effected by the recognized utterance. Reading the wrong stock quote or calendar entry will make the user wait a few seconds, but hanging up or sending a confidential message to the wrong person by mistake could have serious consequences.

16.4.4.3 Insertion Errors

These recognition errors typically occur because of background noise. The illusory utterance will either be rejected or mistaken for an actual command; in either case, the previous methods can be applied. The real challenge is to prevent insertion errors. One option is to provide users with a keypad command to turn off the speech recognizer to talk to someone, sneeze, or simply gather their thoughts. Pressing the keypad command again can restart the recognizer with a simple prompt, such as "What now?" to indicate that the recognizer is listening again.

Whatever the type of error, a general technique for avoiding errors in the first place is to filter recognition results for unlikely user input. For example, a scheduling application might assume that an error has occurred if the user appears to want to schedule a meeting for 3 A.M.

16.4.5 Correction Strategies

If errors do occur, it is important to provide a means for the user to correct the error (assuming they notice it). Flexible correction mechanisms that allow a user to correct a portion of the input are helpful. For example, if the user asks for a weather forecast for Boston for Tuesday, the system might respond, "Thursday's weather for Boston is…." A flexible correction mechanism would allow the user to just correct the day: "No, I said Tuesday." This however is challenging for the system as it would require it to keep the context of the previous query and recognize that Tuesday is a replacement for Thursday.

When possible, using an alternate form of input can alleviate the user's frustration. For dictation systems, eye tracking to rapidly select from a list of alternate words has been shown to be successful (Vertanen and MacKay 2010). For transactional telephony systems, the designer can fall back to telephone keypad with a limited set of choices. If the user is at a prompt where only a few choices are available and the user has encountered several rejection errors, the user could be instructed: "Press any key when you hear the option you want." The telephone keypad also works well when the requested input is numeric (e.g., telephone numbers, account, or social security numbers). Getting users to type alphabetic text using a telephone keypad is not a good idea, and it is to avoid this type of input that speech systems are usually recommended in the first place.

Another strategy is to have the system take its best guess at the requested function. This is a good tactic to take when the number of functions enabled at a particular branch in the dialog is too large to list for the user. A reasonable prompt is "I think you are trying to create a message, is that correct?" If the user answers in the affirmative, the conversation moves forward and the system can present the user with a sample valid utterance for that function. However, if the best guess is wrong, it is not a good idea to keep iterating through the N-best choices because this only leads to user frustration. If the response to the best guess is negative, a better solution is to reprompt with a restricted set of choices. Be sure to eliminate the choice that is definitely wrong (e.g., creating a message in the previous example). The goal is to move away from a very general prompt such as, "I'm sorry I do not understand, please try again" toward a directed prompt that will increase the likelihood of success. A series of errors in a row is a clear indication that simply having the user repeat the utterance, or rephrase it, is not working.

The best guess tactic can be combined with another correction strategy, switching to more constrained grammar, to increase its likelihood of success. For example, in the prior prompt, a directive of what utterances are available to the user can be added: "I think you are trying to create a message, is that correct? Please say yes or no."

16.4.6 Testing and Iterating

Once the preliminary application design is complete, a wizard-of-oz study can help test and refine the interface. The speech data collected from this study can be used to help refine the grammar (Rudnicky 1995) or, in the case of an NLU system, to statistically train the engine on potential utterances. In these studies, a human wizard—often using software tools—simulates the speech interface. Major usability problems are often uncovered with these types of simulations.

A wizard-of-oz study usually involves bringing participants into a lab and telling them they will be interacting with a computer. If it is a telephony application, they can be asked to call a telephone number, at which point a human "wizard" picks up the phone and manipulates software so that recordings of synthesized speech are played to the participant. As the participant makes requests to the computer, the wizard carries out the operations and has the computer speak the responses. Often, none of the participants suspect that they are not interacting with a real speech system (Dahlback, Jonsson, and Ahrenberg 1993). Because computer tools are usually necessary to carry out a convincing simulation, wizard-of-oz studies are more time-consuming and complicated to run than natural dialog studies. If a prototype of the final application can be built quickly, it may be more cost-effective to move directly to a usability study.

With speech applications, usability studies are particularly important for uncovering problems because of recognition errors, which are difficult to simulate effectively in a wizard-of-oz study, but are a leading cause of usability problems. The effectiveness of an application's error recovery functionality must be tested in the environments in which real users will use the application. Conducting usability tests of speech applications can be a bit tricky. Two standard techniques used in tests of graphical applications—facilitated discussions and speak-aloud protocols—cannot be used effectively with speech applications. It is best to have study participants work in isolation, speaking only into a telephone or microphone. A tester should not intervene unless the participant becomes completely stuck. A follow-up interview can be used to collect the participant's comments and reactions. Ultimately, the system needs to be tested with real users, accomplishing real tasks in a realistic environment. This type of testing will provide data that will allow the designer to modify the grammar and prompts as necessary.

16.5 DESIGNING LANGUAGE AND TRANSLATION APPLICATIONS

In light of the many odd jokes about errors made by machine translation, the current justification for using machine translation applications is as a "gisting" tool (so-called Fully Automated Useful Translation) that provides reasonable or useful real-time translation (instead of the more accurate Fully Automated High Quality Translation) to increase productivity and collaboration. The majority of translation applications can be classified into two categories:

1. Applications for viewing or consuming information that exist in a foreign language (informational applications): This includes web page translation, text translation, and speech transcription (or translation) that enable a user to view or read the content of a web page, or e-mail, or call details (excluding the images) in his or her language.

2. Applications for cross-lingual real-time communication or collaboration (collaboration applications): A good example of this involves instant messaging translation which enables a user to engage in a real-time chat with another person, using their respective native languages.

16.5.1 Inherent Challenges of Machine Translation

Similar to what has been discussed in Section 16.4 for speech applications, designing machine translation systems also poses substantial challenges. Understanding these challenges and assessing the various trade-offs that must be made during the design process will also help to produce the most effective interface for machine translation applications.

In our discussion of how machine translation works, we stated that one of the defining characteristics that underscores the usability limitations of the technology is the challenge for the system to know all the relevant words or sentences, along with their meaning in a source language, and then map these to the corresponding words, sentences, and their respective meanings in a target language. This is especially important in collaboration applications where machine translation is used for real-time conversational or translational interactions that requires a user to first perform a task in the source language (native language), which is then automatically rendered in the target language. Therefore, the successful outcome of the translation into the target language is dependent on the accuracy or proper characterization from the source language. What we find is that a lot of errors which require HCI intervention are introduced in this process of transitioning the linguistic forms between the two languages through their associated technological components. We will go into more details on this issue in our discussion of the design process (e.g., handling errors).

Another important consideration is that translation does not only need to deal with the formalism of mapping words between two languages but also finding appropriate ways of expressing cultural meanings in the process. Cultural meanings are intricately woven into the fabric of language. Although current algorithms and models have found ways to systematically deal with the purely observable linguistic aspect of language (mapping of words, phrase, sentences), there is a huge vacuum in mapping cultural meanings as part of the translation process in an effective and systematic way.

Unlike speech applications in which a human interacts with a computer that has been trained to respond with predefined prompts (using various conversational and prompting strategies), machine translation collaborative applications are the *invisible* mediator of the conversational interaction between two humans who speak different languages. Given this characterization, there is an additional complexity in that a different set of emotions (e.g., tolerance for errors, levels of expectation, etc.) apply to speech applications when compared with machine translation applications. The former involves HCI, whereas the latter involve both human–human computer mediation features as well as HCI. This difference is not trivial and needs to be accommodated in the design of machine translation applications.

Based on the issues outlined so far, the inherent challenges in designing machine translation systems (e.g., collaboration applications) can be summed up as follows in Table 16.2.

These design issues in a machine translation application pose greater challenges than those for a speech application and must be addressed in order to have an optimal interface.

16.5.2 Habitability

Habitability refers to the ability of users to stay within the limits of a system's domain while expressing themselves conversationally and productively in the ensuing dialog turns

TABLE 16.2
Summary of the Substantial Challenges Involved in the Design of a Machine Translation Application

Challenges	Speech Interfaces	Translation Interfaces
Recognition errors	Speech recognition errors	Speech recognition errors (for S2S systems); Text recognition errors (T2T systems)
Mapping errors	Map a form to a predefined semantic class (for NLU systems) in a single (shared) language	Map forms between two languages
Human factors errors	HCI	HCI; human–human and computer mediation
Cultural errors	Little or no cultural errors since communication occurs in a single (shared) language	Cultural errors abound; lacks any formalism for mapping cultural meanings between the languages

HCI = human–computer interaction; NLU = natural language understanding

(Watt 1968, Ogden 1988). In a speech application, this can be achieved and augmented by providing help to a user who takes too long to speak or says something that is out-of-grammar. Collaborative machine translation applications, as summarized in Table 16.2, involve human–human interaction that is mediated by a computer. Typically, when people engage in a conversation they seek clarification directly from each other, and because the computer is a mediator rather than a participant, there is really no basis for asking the system to provide help like "What are my choices?" or "What can I say?" Consequently, the design challenge is that of habitability, that is, to ensure that a user stays within the limits of the domain. Unfortunately, most of the current applications do not provide any formal functions in the interface where a user can ask for help on what to say in order to stay within the application's domain. Instead, this is left to the participants in the collaborative exchange to figure it out.

16.5.3 Providing Help

Rules for turn-taking have been formalized for human–human interaction (Sacks, Schegloff, and Jefferson 1974). Thus, one basic user interface design consideration is how a system yields control to the user in the conversational interaction. The issue is not whether turn-taking rules will be violated but rather what sort of help is available to enable the user recover. Only one of the two basic types of help discussed in speech application is relevant in collaborative machine translation applications, and this has to do with system-initiated help. An example is when a user is providing credit card information in a speech-to-speech application. This is one good example of an instance where the tolerance for machine translation errors is usually very low. Therefore, it is important to provide help. The way it works is that upon detecting digits in the user's utterance (especially after repeated attempts), the system can provide context-sensitive help to the user saying "Oh, I see you are trying to provide some numbers, you can do this error-free by simply typing them into the text box, rather than speaking." Another example is in text-to-text systems where the system determines that a particular translation has been mapped with very low confidence score. In this situation, rather than sending the translated text on to the other person, the system informs the user to preview the message by using "back translation," which translates the text back into the original message to allow this user make an informed decision either to simplify the text, or use an entirely different strategy altogether (like using only nouns or action words) to force a better translation.

16.5.4 Handling Errors

Like speech applications, machine translation systems are imperfect (and probably will always be) as the interaction is complicated by two potential sources of errors: recognition and translation. The same three categories of recognition errors (rejection, substitution, and insertion) that were earlier described for speech applications can also be found in the speech component of speech-to-speech machine translation applications. However, unlike speech applications, speech-to-speech machine translation applications do not currently have any systematic way of handling these errors in the interface. As we stated in Section 16.3.7, this is a consequence of how machine translation works. Translation depends on the accuracy of the input (in this case speech), but there is no provision in the interface to handle any of the recognition errors whatsoever, rather, the output from the recognition component is passed on directly to the translation component. For future speech-to-speech applications, this is one area in which the strategies already in place for speech applications can be applied or customized. The errors contained in the output from the translation component can also be modeled after the three types of errors in speech application: rejection, substitution, and insertion. Accordingly, a rejection error occurs when the machine translation algorithm has no valid hypothesis about what the user said. For example, a user says "May I run inside and give the keys to my visitor" and the translation outputs only function words like "in the to my" which is complete gibberish. A substitution error involves the machine translation algorithm changing the entire meaning of the original utterance by mistaking the user's utterance for a different valid utterance. For example, a user's initial message "How are you?" is translated as "How old are you?" which, though valid, sets the conversation on an entirely different path (and because two separate cultures are involved, this may even come across as too direct or offensive). In the case of an insertion error, the machine translation algorithm may decode multiple words when only one was spoken. For example, a user says "hello" but the system says "help me get all." The same vacuum in the handling of recognition errors also exist in translation errors; there are currently no formal or systematic ways in the interface for dealing with these categories of translation errors. As a result, the users are left to interpret errors and then try to repair them by picking from any number of available strategies that they think might work. This creates inconsistency in the interface and impedes the ability to learn how to use the system (learnability) as well as the usability of collaborative machine translation systems in general.

In light of the foregoing discussion of the various gaps in handling errors, we propose that a discourse manager should be included in future user interfaces of collaborative machine translation applications. This component will function much like the dialog manager in speech applications and regulate the various strategies for handling recognition and translation errors. The discourse manager will stand between the ASR and the translation component. In this new approach, the output from the ASR is first sent to the discourse manager which will respond with an appropriate error message to the user, where necessary, instead of passing on the erroneous utterance directly to the translation component. Conversely, the output of the translation manager will pass through the discourse manager to regulate any rejection, substitution, or insertion errors, with an appropriate message to the user, instead of passing on gibberish or utterances with low confidence. In addition, the discourse manager will contain formalism for mapping the various cultural (idiosyncratic) aspects of translation including taboo words, slangs, gender, euphemisms, metaphors, clichés, idioms, innuendos, honorifics, and so on. We believe the introduction of a discourse manager into the process of how machine translation works will empower users and ensure a more optimal interface that is able to guard against (or mitigate) instances of being "lost in translation" while using machine translation applications.

16.5.5 Correction Strategies

Although there are currently no systematic strategies for handling the specific types of errors discussed in Section 16.5.4, there are, however, some other general error correction strategies that provide a means for the user to correct a translation error (not recognition errors because those are passed on automatically from the ASR to the translation component).

In speech-to-speech application, the interface allows users to see the translated message that is passed on to the other participant. The standard practice is to show pictures or images on the interface to be used in disambiguating (or augmenting) a poor translation. For example, based on an English–Chinese speech-to-speech application, if a user says "I want to visit the Forbidden City" but all the words except the location (Forbidden City) is correctly recognized and translated, then the system automatically shows a picture of tourists visiting the Forbidden City to augment the poor translation. Similarly, when an ambiguous word like "duck" is the only recognized and correctly translated word in a sentence "I would like to order a duck," and then showing the user the images of a person in the act of ducking (from an object) and that of a duck (bird), usually helps with disambiguation because the user can simply click on the correct image which then sends a predefined relevant utterance in the target language.

In a text-to-text application, the interface allows the user to be able to suggest a better translation in place of a poor translation from the system. In this regard, the function to "suggest a better translation" is a correction feature that allows users to be able to make corrections from the interface. The drawback or weakness of this feature is that its effectiveness (or use) is based on a user being sufficiently bilingual in both the source and target languages. The way it works is that when a user sees a poorly translated text (in the target language), they can make changes to the text, which is then automatically refreshed in the translation memory. This is not a real-time change, that is, you cannot benefit immediately from your own correction; however, it takes effect in near real time in the sense that the correction is available to the very next user.

Another correction strategy used in machine translation applications is to automatically set the maximum number of re-tries in order to prevent dialog loops. This feature keeps track of successive and repeated errors so that when a predetermined threshold is reached, it informs the user either to try a default strategy (like providing credit card numbers while using speech as input and then being asked to default to text input), or to inform the user to pick from a selection of canned phrases or expressions. The canned list usually contains the most frequently used phrases, expressions, or sentences in the relevant domain. The user goes through the list and simply clicks (selects) the one that closely describes the original message he or she was trying to communicate.

16.5.6 Crowdsourcing and Machine Translation

Crowdsourcing is generally described as a web-based activity that harnesses the creative contributions of a diverse large network of individuals (the crowd) through an open call requesting for their participation and contributions (Howe 2006, 2008; Surowiecki 2004, 2005). The typical crowdsourcing ecosystem is one in which a problem (or task) is posted online (by a company or an individual), and a large number of people (the crowd) are motivated (incented) to solve the problem and receive appropriate rewards (Brabham 2008b). As an emerging web-based mass collaboration strategy, there is a lack of formal generalizations about the crowdsourcing community at large because there are currently so many different examples and uses that include www.youtube.com (videos and pictures), www.innocentive.com (basic level research and development), www.sringwise.com (recognizing upcoming trends and weak signals), www.threadless.com (end-product design), www.iStockphoto.com (pictures), www.amazon.com (product recommendations and ratings), and so on. Based on these examples, the scope of crowdsourcing appears to be very diverse even as the application of the strategy continues to expand into new areas (or uses). As far as we are aware, there is very little research on formalizing the differences in crowd behavior with respect to the nature of the task (Whittaker et al. 1998, Viitamaki 2008, Stewart et al. 2010). Given its popularity and exponential growth, it is important to have a model or characterization of crowdsourcing that goes beyond listing various types or characteristics (see Table 16.3). Viitamaki (2008) proposes an initial formalization of the types of crowdsourcing communities based on

TABLE 16.3
Typology of Crowdsourcing Based on Community and Orientation Features

Characteristics	Example
No strong community or community interaction between participants	Innocentive Amazon's mTurk
Individuals create in explicit competition; community interacts and, for example, helps select the best material	Threadless Ideastorm
Community cocreates and cooperates in a joint venture without financial commitment	Cambrian-House Wikipedia
Community makes financial investments and has a large role in directing action. Community "owns" the initiative	Sellaband MyFootballClub

the characteristics derived from community features and time orientation.

Although this proposal is foundational, however, as can be observed, the characteristics applied are a mix of marketing, social networking behavior, and financial. These are simply general and descriptive attributes that say nothing about the nature of the task. In our view, it is the nature of the task that defines a crowdsourcing community and pulls a crowd together (i.e., the essence of crowdsourcing). Therefore, based on the nature of the task, Stewart et al. (2010) propose that there are three general kinds of crowdsourcing communities (Table 16.4).

This classification of crowdsourcing allows us to systematically analyze various crowdsourcing communities. For example, focusing on the individualistic category we observe underlying behavioral similarities between the iStockphoto crowdsourcing community (Brabham 2008a) and the n.Fluent language translation crowdsourcing community (Stewart et al. 2009b) in that they are both individualistic. In both communities, the task requires that participants are one-person units and there is no need for social networking or collaboration among them. They are not interested in building a network of friends or creative professionals.

Crowdsourcing has become very central in the implementation of machine translation applications. In this context, it is often referred to as “community translation.” As the Internet grows in popularity and adoption, so has the use of online machine translation to make information from search portals consumable by the global community. In fact, millions of people turn to www.google.translate.com and other translation portals everyday seeking a cheap (no cost), fast, and reasonable translation of e-mails, text messages, and web pages. In general, this service is free to the end-users who, by using it, provide the much needed data (parallel corpora) for enhancing the SMT engines. This is the background for the two areas where crowdsourcing is currently used in machine translation. First, it is used as an error recovery (correction) strategy for improving imperfect translations. As previously discussed in Section 16.5.4 on handling errors, the translation interface allows (encourages) the user (who is bilingual) to make corrections to the translated content. Based on this strategy, improvements can be made to the translation memory as the crowd provides corrections to poorly translated texts, which are then used to update the translation memory for providing better translations. Another important use of crowdsourcing in machine translation is for data collection (parallel corpora) which is required for improving the SMT algorithms. As a case in point, Stewart et al. (2009b) describes the use of crowdsourcing inside IBM for collecting data to improve its SMT engines by creating a crowdsourcing community with an open call to about 400,000 employees spread over 160 countries. The crowdsourcing platform successfully harnesses the linguistic skills of bilingual employees in translating sentences from English to their native language (or vice-versa), for 11 language pairs including Arabic, Portuguese (Brazilian), Chinese (traditional), Chinese (simplified), French, Italian, Japanese, Korean, Portuguese, Spanish, and Russian.

TABLE 16.4
Typology of Crowdsourcing Based on Nature of the Task

Type	Characteristics and Example
Collectivistic	This involves a task where several people are handling small parts of a larger problem (e.g., uTest), and the community exhibits the need to collaborate and network with each other using available social tools.
Individualistic	This involves a task where many people are contributing (individually) toward a single goal or task (e.g., iStockphoto), and do not need to “socialize” with each other.
Collectivistic-individualistic	This is a mixture of collectivistic and individualistic properties, wherein the crowd is made up of subgroups who cooperate or collaborate to compete against other subgroups in the quest of completing a task (e.g., 2009 Defense Advanced Research Projects Agency experiment where 10 balloons were placed across the United States and teams were challenged to compete to be the first to report the location of all the balloons.

16.6 CONCLUSION

The design and development lifecycle for an effective speech and language translation application differs from a traditional GUI application. An effective speech and language translation application is one that uses speech and/or translation to enhance a user’s performance of a task or enable an activity that cannot be done without it. However, the design process does share common elements with the design of a successful GUI application, such as the need to first understand the task that the users are trying to accomplish. In the case of speech, modeling the task can usually be accomplished by listening to how humans accomplish the task today. Sometimes this involves listening to the interactions that take place between a call center employee and a caller; sometimes it involves constructing a situation between two humans and asking them to converse to accomplish the task. Observing users during the task modeling phase helps a designer to understand who the users are and what their goals are. Listening carefully to users while conducting natural dialog studies shows how they speak in the context of the task. A natural dialog study also ensures that prompts and feedback follow the conversational conventions that users expect in a cooperative interaction.

Once the task is modeled and well understood, and the business goals have been defined, the designer/developer must select the appropriate technology for the task, users, and context of use. Conversational interfaces that use speech technologies capitalize on human expertise in interacting with each other through the use of language. Not all tasks will be able to use NLU given the resource requirements and the need for a constrained domain. However, even speech applications that are built with grammar can carry many of the usability advantages of NLU and have a conversational feel to them if the prompts are carefully crafted. Successful dialogs will move between user-initiated, system-initiated, and mixed-initiative styles depending on the state of the task and the history of the interaction.

The HCI design issues in a machine translation application pose greater challenges than those for speech applications summarized in the previous paragraph. Language remains one of the major barriers in a linguistically diverse and globally connected world, with an ever-increasing need for people to communicate and collaborate with each other. There are research challenges to solve in this area before successful language applications are realized in real-world deployment of telephone or Internet-based (non face-to-face) speech-to-speech translation systems. With the exception of the multilingual automatic speech-to-speech translator system, face-to-face, speech-to-speech translation systems use devices like smartphones. Currently, there are many successful implementations of text-to-text translation systems, especially those that are offered as widgets on web pages on the Internet and various online software programs. To move beyond these speech applications to speech-to-speech translation systems, HCI research is needed to design interventions and mitigate issues introduced by the process of transitioning the linguistic constructs between the two languages, and expressing cultural meanings. These challenges stem from the complexity of managing and designing for both human–human computer mediation features as well as HCI in language translation applications.

Beyond the strategies described throughout this chapter, another key to successful design of usable and effective speech and language translation user interfaces is to follow an iterative HCI process. Not even the most experienced designer can craft a perfect dialog in the first iteration. Once an application is designed, wizard-of-oz and usability studies provide opportunities to test interaction techniques and refine application behavior based on feedback from prototypical users. To ensure that the system will be used over time, the designer must modify the design as new data is collected. One of the critical design aspects is enabling users to gracefully recover from errors in speech and language translation applications. Error recovery must be carefully designed for successful user acceptance and use of an application.

Finally, testing the design with target users ensures that the prompts are clear, that feedback is appropriate, and that errors are caught and corrected. Also, as part of the testing, verifying that the design accomplishes the business goals that were set out to be achieved helps the application gain approval from the sponsors. If problems are uncovered during testing, the design should be revised and tested again. By focusing on users and iterating on the design, one can produce an effective, polished speech interface design.

Looking to the future in speech and language translation research and development, we think that there will be increased focus on these applications. In terms of speech applications, continued research is needed to create more adventurous and successful dialogs that balance user-initiated, system-initiated, and mixed-initiative styles depending on the user's goal and context of use. In terms of language translation applications, the Internet has truly made the world a global village as it has removed or mitigated the traditional space and time constraints from collaboration. The next challenge is to make communication in local languages possible in order to empower people to engage in cross-national, cross-lingual collaboration. The design and development of advanced speech applications and robust speech-to-speech translations systems has the potential to create new capabilities of high value to users in both business and personal settings. The incorporation of HCI methods in these research and development efforts is critical in increasing the probability of success and decreasing the time and effort to create successful applications for users around the world.

REFERENCES

Aykin, N. 2005. *Usability and Internationalization of Information Technology*. 4–19. Princeton, NJ: Lawrence Erlbaum Associates.

Ballentine, B., and D. Morgan. 1999. *How to Build a Speech Recognition Application: A Style Guide for Telephony Dialogs*. San Ramon, CA: Enterprise Integration Group, Inc.

Brabham, D. C. 2008a. Moving the crowd at IStockphoto: The composition of the crowd and motivations for participation in crowdsourcing application. *First Monday* 13:6.

Brabham, D. C. 2008b. Crowdsourcing as a model for problem solving: an introduction and cases. *Convergence: Int J Res New Media Technol* 14(1):75–90.

Chai, J. 2001. Natural Language Sales Assistant—a Web-based Dialog System for Online Sales. To be presented at the Thirteenth Conference on Innovative Applications of Artificial Intelligence.

Clark, H. 1993. *Arenas of Language Use*. Chicago, IL: University of Chicago Press.

Cohen, M., J. Giangola, and J. Balogh. 2004. *Voice user Interface Design*. Boston, MA: Addison-Wesley.

Dahlback, N., A. Jonsson, and L. Ahrenberg. 1993. Wizard of oz studies—why and how. In *IUI '93 Proceedings of the 1st international conference on intelligent user interfaces*, 193–200. New York: ACM.

Feng, J., A. Sears, and C. M. Karat. 2006. A longitudinal evaluation of hands-free speech-based navigation during dictation. *Int J Hum Comput Stud* 64(6).

Francis, A., and H. Nusbaum. 1999. Evaluating the quality of synthetic speech. In *Human Factors and Voice Interactive Systems*, ed. D. Gardner-Bonneau, Boston, MA: Kluwer Academic Publishers.

Gao, Y., B. Zhou, Z. Diao, J. Sorensen, and M. Picheny. 2002. "MARS: A statistical semantic parsing and generation-based multilingual automatic translation system." *J Mach Transl* 17:185–212.

Gong, L., and J. Lai. 2003. To mix or not to mix synthetic speech and human speech: Contrasting impact on judge-rated task. *Int J Speech Technol* 6(2):123–31.

Goshawke, W., I. D. K. Kelly, and J. D. Wigg. 1987. *Computer Translation of Natural Language*. Wilmslow, UK: Sigma Press.

Halverson, C. A., D. A. Horn, C. Karat, and J. Karat. 1999. The beauty of errors: Patterns of error correction in desktop speech systems. In *Human-Computer Interaction—INTERACT '99*, ed. M. A. Sasse and C. Johnson, 133–40. Amsterdam: IOS Press.

Howe, J. 2006. The rise of crowdsourcing. *Wired* 14(6), http://www.wired.com/wired/archive/14.06/crowds.html (accessed June 12, 2006).

Howe, J. 2008. *Crowdsourcing: Why the Power of the Crowd is driving the Future of Business*. New York: Random House Publishers.

Kamm, C. 1994. User interfaces for voice applications. In *Voice Communication Between Humans and Machines*, ed. D. B. Roe and J. G. Wilpon, 422–444. Washington, DC: National Academy Press.

Karat, C., C. Halverson, D. Horn, and J. Karat. 1999. Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Human Factors in Computing Systems—CHI 99 Conference Proceedings*, ed. M. Altom and M. Williams, 568–575. New York, NY: ACM.

Karat, C., and J. Karat. 2010. Designing and evaluating usable technology in industrial research: Case studies in speech recognition, personalization in ecommerce, and security and privacy. In *Lectures in Human-Centered Informatics*, ed. J. Carroll, New York, NY: Morgan-Claypool.

Karat, J., D. Horn, C. Halverson, and C. Karat. 2000. Overcoming un-usability: Developing efficient strategies in speech recognition systems. *CHI '00 extended abstracts on Human factors in computing systems*, 141–2. New York: ACM.

Karat, J., J. Lai, C. Danis, and C. Wolf. 1999. Speech user interface evolution. In *Human Factors and Voice Interactive Systems*, ed. D. Gardner-Bonneau, 1–35. Boston, MA: Kluwer.

Lai, J., D. Wood, and M. Considine. 2000. The effect of task conditions on the comprehensibility of speech. In *CHI '2000 Conference on Human Factors in Computing Systems*, New York: ACM.

Ogden, W. C. 1988. Using natural language interfaces. In *Handbook of Human-Computer Interaction*, ed. M. Helander, 281–99. North-Holland, Amsterdam: Elsevier.

Price, K. J., M. Lin, J. Feng, R. Goldman, A. Sears, and J. A. Jacko. 2006. Motion does matter: An examination of speech-based text entry on the move. *Universal Access in the Information Society* 4(3):246–257.

Price, K., and A. Sears. 2005. Speech-based text entry for mobile handheld devices: An analysis of efficacy and error correction techniques for server-based solutions. *Int J Hum Comput Interact* 19(3):279–304.

Rabiner, L. R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of IEEE* 77:257–86. New Jersey: IEEE.

Roe, D. B., and J. G. Wilpon. 1993. Wither speech recognition: The next 25 years. *IEEE Commun Mag* 11:54–62.

Rudnicky, A. 1995. The design of spoken language interfaces. In *Applied Speech Technology*, ed. A. Syrdal, R. Bennett, and S. Greenspan, Boca Raton, FL: CRC Press.

Sacks, H., E. Schegloff, and G. Jefferson. 1974. A simplest systematics for the organization of turntaking for conversation. *Language* 50(4):696–735.

Schmandt, C. 1994. *Voice Communication with Computers: Conversational Systems*. New York, NY: Van Nostrand Reinhold.

Sears, A., C. Karat, K. Oseitutu, A. Kaimullah, and J. Feng. 2000. Productivity, satisfaction, and interaction strategies of individuals with spinal cord injuries and traditional users interacting with speech recognition software. *Universal Access in the Inf Soc* 1:5–25.

Stewart, O., J. Huerta, M. Sader, A. Sakrajda, J. Marcotte, and D. Lubensky. 2009b. Designing crowdsourcing for the enterprise. In *SIG KDD HCOMP Workshop.* New York: ACM.

Stewart, O., D. Lubensky, J. Huerta, J. Marcotte, C. Wu, and A. Sakrajda. 2010. Crowdsourcing participation inequality: A SCOUT model for the enterprise domain. In *SIG KDD HCOMP Workshop.* New York: ACM.

Stewart, O., M. Picheny, D. Lubensky, and B. Ramabhadran. 2009a. Cultural voice markers in speech-to-speech machine translation systems. *Proceedings of the International Workshop on Intercultural Collaboration*. New York: ACM.

Surowiecki, J. 2004. *The Wisdom of the Crowds: Why the many are Smarter than the few and how Collective Wisdom Shapes Businesses, Economies, Societies, and Nations*. New York: Doubleday.

Surowiecki, J. 2005. *The Wisdom of Crowds*. New York: Doubleday.

TAUS Report. 2007. TAUS (Translation Automation User Society) Starter's Guide to Machine Translation: Technologies, Case Studies & Good Practices. Release 1: April 2007.

Vertanen, K., and D. J. C. MacKay. 2010. Speech dasher: Fast writing using speech and gaze; In *CHI '10 Proceedings of the 28th international conference on Human factors in computing systems*, 595–8. New York: ACM.

Viitamaki, S. 2008. The FLIRT model of crowdsourcing: planning and executing collective customer collaboration. MA thesis: Helsinki School of Economics.

Watt, W. C. 1968. Habitability, *Am Doc* 19(3):338–51.

Weinschenk, S., and D. T. Barker. 2000. *Designing Effective Speech User Interfaces*. New York, NY: Wiley.

Whittaker, S., L. Terveen, W. C. Hill, and L. Cherny. 1998. The dynamics of mass interaction. In *Proceedings of ACM Conference on Computer-supported Cooperative Work, CSCW '98*, 257–64. New York: ACM.

Yankelovich, N. 1996. How do users know what to say? *Interactions* 3(6):32–43.

Yankelovich, N. 2008. "Using natural dialogs as the basis for speech interface design". In *Human Factors and Voice Interactive Systems*, ed. D. Gardner-Bonneau and H. E. Blanchard, 2nd ed., 255–90. New York: Springer Science+Business Media, LLC.

Yankelovich, N., G. A. Levow, and M. Marx. 1995. Designing SpeechActs: Issues in speech user interfaces. In *CHI '95 Proceedings of the SIGCHI conference on Human factors in computing systems*, 369–76. New York: ACM Press/Addison-Wesley.

Zhou, B., D. Dechelotte, and Y. Gao. 2004. "Two-way speech-to-speech translation on handheld devices." In *Int. Conf. of Spoken Language Processing (ICSLP)*, 1637–1640. http://www.isca-speech.org/archive.

17 Multimedia User Interface Design

Alistair Sutcliffe

CONTENTS

17.1 INTRODUCTION

The distinguishing characteristics of multimedia are information-intensive applications that have a complex design space for presenting information to people. Design of multimedia interfaces currently leaves a lot to be desired. As with many technologies, it is the fascination with new devices, functions, and forms of interaction that has motivated design rather than ease of use, or even utility of practical applications. Poor usability limits the effectiveness of multimedia products, which might look good but do not deliver effective use (Scaife et al. 1997). With the growth of the web, use of media has become a vital component of attractive and engaging design.

This chapter describes a design process that starts with an information analysis then progresses to deal with issues of media selection and integration. The background to the method and its evolution with experience can be found in several publications (Sutcliffe and De Angeli 2005; Faraday and Sutcliffe 1996, 1997, 1998; Sutcliffe and Faraday 1994). A more detailed description is given in Sutcliffe (2003). The time-to-market pressure gives little incentive for systematic, principled design, so at first reading, a systematic approach may seem to be counter to the commercial drivers of development. However, I would argue that if multimedia design does not adopt a usability engineering approach, it will fail to deliver effective and usable products.

Traditional multimedia markets have been in education and training, although dialogue in many systems has been restricted to drill and quiz interaction, interactive simulations and microworlds are more effective (Rogers and Scaife 1998). Multimedia has been used extensively in task-based applications in process control and safety-critical systems (Alty 1991; Hollan, Hutchins, and Weitzman 1984). With the advent of the Web 2.0 and beyond, interactive multimedia is a continuing design challenge.

Design for multimedia user interfaces (UIs) expands conventional definitions of usability (e.g., ISO 9241 Part 11: ISO 1997) into five concerns as follows:

1. *Operational usability:* It is the conventional sense of usability that concerns design of graphical user interface (GUI) features such as menus, icons, metaphors, and navigation in hypermedia.
2. *Information delivery:* It is a prime concern for multimedia or any information-intensive application and raises issues of media selection, integration, and design for attention.
3. *Learning*: Training and education are both important markets for multimedia and hence learnability of the product and its content are key quality attributes. However, design of educational technology is a complex subject in its own right, and multimedia is only one part of the design problem.
4. *Utility*: In some applications, this will be the functionality that supports the user's task; in others, information delivery and learning will represent the value perceived by the user.
5. *Engagement and attractiveness*: The attractiveness of multimedia is now a key factor especially for websites. Multimedia interfaces have to attract users and deliver a stimulating user experience, as well as being easy to use and learn.

Multimedia design involves several specialisms, which are technical subjects in their own right. For instance, design of text is the science (or art) of calligraphy that has developed new fonts over many years; visualization design encompasses the creation of images, either drawn or captured as photographs. Design of moving images, cartoons, video, and film are further specializations, as are musical composition and design of sound effects. Multimedia design lies on an interesting cultural boundary between the creative artistic community and science-based engineering. One implication of this cultural collision is that space precludes "within media" design, that is, guidelines for design of one particular medium, being dealt with in depth in this chapter. Successful multimedia design often requires teams of specialists who contribute from their own skill sets (Kristof and Satran 1995; Mullet and Sano 1995).

17.2 DEFINITIONS AND TERMINOLOGY

Multimedia essentially extends the GUI paradigm by providing a richer means of representing information for the user by use of image, video, sound, and speech. The following definitions broadly follow those in the ISO standard 14915 on Multimedia User Interface Design (ISO 1998). The starting point is to ask about the difference between what is perceived by someone and what is stored on a machine.

Communication concepts in multimedia can be separated into the following:

- *Message*: The content of communication between a sender and receiver.
- *Medium (*plural *media)*: The means by which that content is delivered. Note that this is how the message is represented rather than the technology for storing or delivering a message. There is a distinction between perceived media and physical media such as CD-ROM and hard disk.
- *Modality*: The sense by which a message is sent or received by people or machines. This refers to the senses of vision, hearing, touch, smell, and taste.

A message is conveyed by a medium and received through a modality. A modality is the sensory channel that we use to send and receive messages to and from the world, essentially our senses. Two principal modalities are used in human–computer communication as follows:

1. Vision: All information received through our eyes, including text and image-based media
2. Hearing: All information received through our ears, as sound, music, and speech

In the future, as multimedia converges with virtual reality (VR), we will use other modalities more frequently: haptic (sense of touch), kinaesthetic (sense of body posture and balance), gustation (taste), and olfaction (smell). These issues are dealt with in Chapters 19 and 29.

Defining a medium is not simple because it depends on how it was captured in the first place, how it was designed, and how it has been stored. For example, photograph can be taken on film, developed, and then scanned into a computer as a digitized image. The same image may have been captured directly by a digital camera and sent to a computer as an e-mail file. At the physical level, media may be stored by different techniques.

Physical media storage has usability implications for the quality of image and response time in networked multimedia. A screen image with 640 × 480 VGA resolution using 24 bits per pixel for good color coding gives 921,600 bytes, so at 30 frames per second, 1 second needs around 25 megabytes of memory or disk space. Compression algorithms, for example, MPEG (Moving Pictures Expert Group), reduce this by a factor of 10. Improvements in disc storage have reduced file size concerns; however, physical image media constraints are still important on networks, when bandwidth limits the desired display quality. For example, the low-resolution video on mobile phones is typically transmitted at 15 frames per second at a resolution of 240 × 320, although high-resolution cameras 1270 × 780 will be available in the near future. In contrast, Internet video and film on-demand services deliver much higher picture quality, by using intelligent fetch-ahead algorithms, but they need good quality and broadband connections. Sound, in comparison, is less of a problem. Full stereo audio with a complete range of harmonic frequencies consumes only 100 kilobytes for 5 minutes, so there are few technology constraints on high-quality audio.

17.3 COGNITIVE BACKGROUND

The purpose of this section is to give a brief overview of cognitive psychology as it affects multimedia design. More details can be found in Part I, Humans in HCI.

17.3.1 Perception and Comprehension

Generally, our eyes are drawn to moving shapes, then complex, different, and colorful objects. Visual comprehension can be summarized as "what you see depends on what you look at and what you know." Multimedia designers can influence what users look at by controlling attention with display techniques such as use of movement, highlighting, and salient icons. However, designers should be aware that the information people assimilate from an image also depends on their internal motivation, what they want to find, and how well they know the domain (Treisman 1988). A novice will not see interesting plant species in a tropical jungle, whereas a trained botanist will. Selection of visual content, therefore, has to take the user's knowledge and task into account.

Because the visual sense receives information continuously, it gets overwritten in working memory (Baddeley 1986). This means that memorization of visually transmitted information is not always effective unless users are given time to view and comprehend images. Furthermore, users only extract very high-level or "gist" (general sense) information from moving images. Visual information has to be understood by using memory. In realistic images, this process is automatic; however, with nonrealistic images, we have to think carefully about the meaning, for example, to interpret a diagram. Although extraction of information from images is rapid, it does vary according to the complexity of the image and how much we know about the domain. Sound is a transient medium, so unless it is processed quickly, the message can be lost. Even though people are remarkably effective at comprehending spoken language and can interpret other sounds quickly, the audio medium is prone to interference because other sounds can compete with the principal message. Because sound is transient, information in speech will not be assimilated in detail, and so only the gist will be memorized (Gardiner and Christie 1987).

17.3.2 Selective Attention

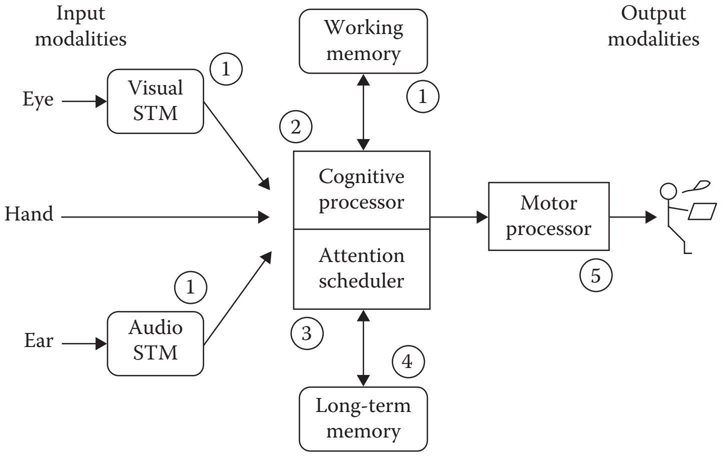
We can only attend to a limited number of inputs at once. Although people are remarkably good at integrating information received by different senses (e.g., watching a film and listening to the soundtrack), there are limits determined by the psychology of human information processing (Wickens, Sandry, and Vidulich 1983). Our attention is selective and closely related to perception; for instance, we can overhear a conversation in a room with many people speaking (the cocktail party effect). Furthermore, selective attention differs between individuals and can be improved by learning: for example, a conductor can distinguish the different instruments in an orchestra, whereas a typical listener cannot. However, all users have cognitive resource limitations, which means that information delivered on different modalities (e.g., by vision and sound) has to compete for the same resource. For instance, both speech and printed text require a language-understanding resource, whereas video and a still image use image interpretation resources. Cognitive models of information-processing architectures (e.g., interacting cognitive subsystems: Barnard 1985) can show that certain media combinations will not result in effective comprehension because they compete for the same cognitive resources, thus creating a processing bottleneck. We have two main perceptual channels for receiving information: vision and hearing; information going into these channels has to be comprehended before it can be used. Figure 17.1 shows the cognitive architecture of human information processing and resource limitations that lead to multimedia usability problems.

Capacity overflow (1) may happen when too much information is presented in a short period, swamping the user's limited working memory, and cognitive processor's capability to comprehend, chunk, and then memorize or use the information. The connotation is to give users control over the pace of information delivery. Integration problems (2) arise when the message on two media is different, making integration in working memory difficult; this leads to the thematic congruence principle. Contention problems (3) are caused by conflicting attention between dynamic media, and when two inputs compete for the same cognitive resources. For example, speech and text require language understanding. Comprehension (4) is related to congruence; we understand the world by making sense of it with our existing long-term memory. Consequently, if multimedia content is unfamiliar, we cannot make sense of it. Finally, multitasking (5) makes further demands on our cognitive processing, so we will experience difficulty in attending to multimedia input while performing output tasks.

Making clear a theme in a multimedia presentation involves directing the user's reading and viewing sequence across different media segments. Video and speech are processed in sequence, whereas text enforces a serial reading order by the syntactic convention of language. In contrast, viewing image media is less predictable since it depends on the size and complexity of the image, the user's knowledge of the contents, task and motivation (Norman and Shallice 1986), and designed effects for salience. Attention-directing effects can increase the probability that the user will attend to an image component, although no guarantee can be given that a component will be perceived or understood.

17.3.3 Emotion and Arousal

The content of image media in particular can evoke an emotional response, which can be used to promote a more exciting and engaging user experience. These issues are dealt with more extensively in other chapters; for example, the use of human image and speech to persuade users. People treat human photographs, video, and even animated characters with similar social responses as they give to real people, so human image content can be used to increase interest and



Bottlenecks
1. Capacity overflow: information overload
2. Integration: common message?
3. Contention: conflicting channels
4. Comprehension
5. Multitasking input/output

STM = short-term memory
Similar to working memory,
limited capacity,
continuously overwritten
with new input

FIGURE 17.1 Approximate model of human information processing using a "human as computer system" analogy, based on the Model Human Processor. For more on cognitive models, see Chapter 2 (Proctor and Vu) and Chapter 5 from the Second Edition of the HCI Handbook (Byrne). (From Card, S. K., T. P. Moran, and A. Newell. *The Psychology of Human Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1983. With permission.)

draw attention. Emotional responses (see Chapter 4 from the Second Edition of the HCI Handbook) can be invoked not only by content but also by surprise interactive effects, for example, when a character suddenly appears to challenge the users. Surprise effects, moving image, and stimulating images we are not expecting, all affect the arousal system that broadly equates with our feeling of excitement. Designs that stimulate our arousal are more likely to be memorable and engaging.

17.3.4 Learning and Memorization

Learning is the prime objective in tutorial multimedia. In these applications, the objective is to create a rich memory schema, which can be accessed easily in the future. We learn more effectively by active problem solving or learning by doing. This approach is at the heart of constructivist learning theory (Papert 1980), which has connotations for tutorial multimedia. Interactive microworlds where users learn by interacting with simulations, or constructing and testing the simulation, give a more vivid experience that forms better memories (Rogers et al. 1998). Multiple viewpoints help to develop rich schemata by presenting different aspects of the same problem, so the whole concept can be integrated from its parts. An example might be to explain the structure of an engine, then how it operates, and finally display a causal model of why it works. Schema integration during memorization fits the separate viewpoints together.

The implications from psychology are summarized in the form of multimedia design principles (ISO, 14915, Part 3 Media Integration, ISO 1997). The principles are high-level concepts, which are useful for general guidance, but they have to be interpreted in a context to give more specific advice.

- *Thematic congruence*: Messages presented in different media should be linked together to form a coherent whole. This helps comprehension as the different parts of the message make sense by fitting together. Congruence is partly a matter of designing the content so it follows a logical theme, for example, the script or story line makes sense and does not assume too much about the user's domain knowledge; and partly a matter of attentional design to help the user follow the message thread across different media.
- *Manageable information loading*: Messages presented in multimedia should be delivered at a pace which is either under the user's control or at a rate that allows for effective assimilation of information without causing fatigue. The rate of information delivery depends on the quantity and complexity of information in the message, the effectiveness of the design in helping the user extract the message from the media, and the user's domain knowledge and motivation. Some ways of reducing information overload are to avoid excessive use of concurrent dynamic media and give the user time to assimilate complex messages.
- *Ensure compatibility with the user's understanding*: Media should be selected that convey the content in a manner compatible with the user's existing knowledge, for example, the radiation symbol and road sign icons are used to convey hazards and dangers

to users who have the appropriate knowledge and cultural background. The user's ability to understand the message is important for designed image media (diagrams, graphs) when interpretation is dependent on the user's knowledge and background.

- *Complementary viewpoints*: Similar aspects of the same subject matter should be presented on different media to create an integrated whole. Showing different aspects of the same object, for example, picture and design diagram of a ship can help memorization by developing richer schema and better memory cues.
- *Consistency*: It helps users learn an interface by making the controls, command names, and layout follow a familiar pattern. People recognize patterns automatically, so operating the interface becomes an automatic skill. Consistent use of media to deliver messages of a specific type can help by cueing users with what to expect.
- *Interaction and engagement*: They help understanding and learning by encouraging the user to problem solve. Memory is an active process. Interaction increases arousal and this make the user's experience more vivid, exciting, and memorable.
- *Reinforce messages*: Redundant communication of the same message on different media can help learning. Presentation of the same or similar aspects of a message helps memorization by the frequency effect. Exposing users to the same thing in a different modality also promotes rich memory cues.

17.4 DESIGN PROCESS

Multimedia design has to address the problems inherent in the design of any UI, namely, defining user requirements, tasks, and dialogue design; however, there are three issues that concern multimedia specifically:

1. *Matching the media to the message* by selecting and integrating media so the user comprehends the information content effectively.
2. *Managing users' attention* so key items in the content are noticed and understood, and the user follows the message thread across several media.
3. *Interaction and navigation* so the user can access, play, and interact with media in an engaging and predictable manner.

Figure 17.2 gives an overview of the design process that addresses these issues.

The method shown in the figure starts by requirements and information analysis to establish the necessary content and communication goals of the application. It then progresses to domain and user characteristic analysis to establish a profile of the user and the system environment. The output from these stages feeds into media selection and integration, which match the information requirements to available media resources. This is interleaved with interaction design unless the application is restricted to information presentation. Design then progresses to thematic integration of the user's reading/viewing sequence and design to direct the users' attention. Even though the process is described as a sequence, in practice, the stages are interleaved and iterated; however, requirements, information modeling, and media selection should be carried out, even if they are not complete, before subsequent design stages commence.

Design approaches in multimedia tend to be iterative and user-centered. *Storyboards* are a well-known means of informal modeling in multimedia design (Nielsen 1995; Sutcliffe 1999). Originating from animation and cartoon design, storyboards are a set of images that represent key steps in a design. Translated into software, storyboards depict key stages in interaction and are used for conducting walk-throughs to explain what happens at each stage. Allowing the users to edit storyboards and giving them a construction kit to build their own encourages active participation. Storyboards are followed by building concept demonstrators using multimedia authoring tools (e.g., Macromedia Director, Adobe Dreamweaver) to rapidly develop early prototypes. *Concept demonstrators* are active simulations that follow a scenario script of interaction; departure from the preset sequence is not allowed. Several variations can be run to support comparison; however, the user experience is passive. In contrast, users can test *interactive prototypes* by running different commands or functions. The degree of interactivity depends on the implementation cost, which increases as prototypes converge with a fully functional product.

17.4.1 Users, Requirements, and Domains

The starting point for multimedia, as in all applications, is requirements analysis. The difference in multimedia lies in the greater emphasis on information requirements. A variety of analytic approaches can be adopted, such as task analysis (see Chapter 43), contextual inquiry (Chapter 44), or scenario analysis (Chapter 48). Requirements are listed and categorized into information, task-related, and nonfunctional classes. These will be expanded in subsequent analyses.

It is important to get a profile of the target user population to guide media selection. There are three motivations for user analysis:

1. *Choice of modalities*: This is important for people with disabilities, but also for user preferences. Some people prefer verbal-linguistic material over image.
2. *Tuning the content*: This is presented to the level of users' existing knowledge. This is particularly important for training and educational applications.
3. *Capturing the users' expectations*: So the experience can be geared to their background, for example, different styles for younger people, older people, culture, and socioeconomic audiences.

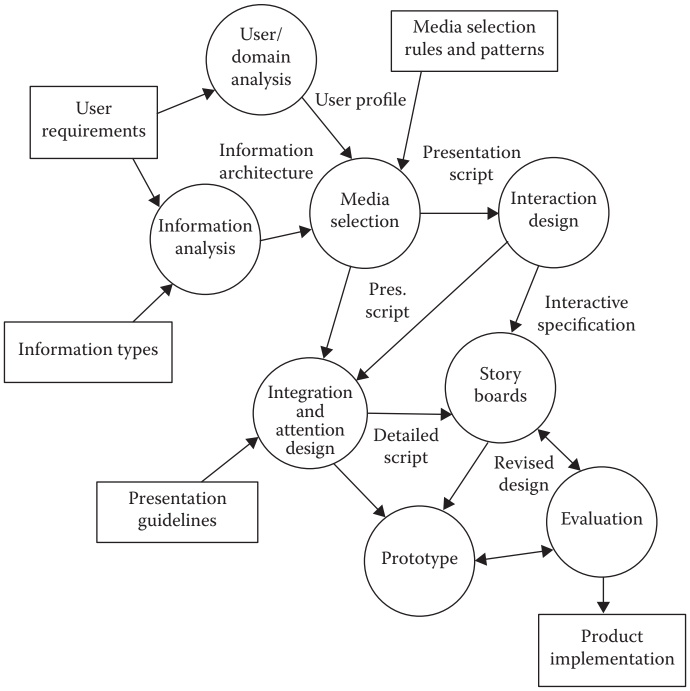


FIGURE 17.2 Overview of the multimedia design process expressed as a data flow diagram.

Acquiring information about the level of experience possessed by the potential user population is important for customization. User profiles are used to design training applications to ensure that the right level of tutorial support is provided, and to assess the users' domain knowledge so that appropriate media can be selected. This is particularly important when symbols, designed images, and diagrams may be involved. The role and background of users will have an important bearing on design. For example, marketing applications will need simple focused content and more aesthetic design, whereas tutorial systems need to deliver detailed content. Information kiosk applications need to provide information, as do task-based applications, but decision-support and persuasive systems (Fogg 1998; see also Chapter 14) also need to ensure that users comprehend and are convinced by messages. Domain knowledge, including use of conventions, symbols, and terminology in the domain, is important because less-experienced users will require more complete information to be presented.

The context and environment of a system will also have an important bearing on design. For example, tourist information systems in outdoor public areas will experience a wide range of lighting conditions, which can make image and text hard to read. High levels of ambient noise in public places or factory floors can make audio and speech useless. Hence, it is important to gather information on the location of use (office, factory floor, public/private space, and hazardous locations), pertinent environmental variables (ambient light, noise levels, and temperature), usage conditions (single user, shared use, broadcast), and expected range of locations (countries, languages, and cultures). Choice of language, icon conventions, interpretation of diagrams, and choice of content all have a bearing on design of international UIs.

As well as gathering general information about the system's context of use, domain modeling can prove useful for creating the system metaphor. Domain models are recorded as sketches of the work environment showing the layout and location of significant objects and artifacts, accompanied by lists of environmental factors. Structural metaphors for organizing information and operational metaphors for controls and devices have their origins in domain analysis.

17.4.2 Information Architecture

This activity consists of several activities that will differ according to the type of application. Some applications might have a strong task model, for instance, a multimedia process control application where the tasks are monitoring a chemical plant, diagnosing problems, and supporting the operator in controlling plant operation. In task-driven applications, information requirements are derived from the task model. In information-provision applications, such as websites with an informative role, information analysis involves categorization and the architecture generally follows a hierarchical model. In the third class of explanatory or thematic applications, analysis is concerned with the story or argument, that is, how the information should be explained or delivered. Educational multimedia and websites with persuasive missions fall into the last category.

In task-driven applications, information needs are annotated on to the task model following a walk-through asking what information the users need to complete the task subgoal, or to take a decision at this step, or to provide as input (see Sutcliffe [1997] for more detail). In information-provision applications, classification of the content according to one of more user views defines the information architecture; for example, most university departments have an information structure with upper-level categories for research, undergraduate courses, postgraduate courses, staff interests, departmental organization, mission and objectives, and so on. For explanatory applications, a theme or story line needs to be developed. This will depend on the application's objectives and the message the owner wishes to deliver. An example thematic map from a health awareness application is illustrated in Figure 17.3.

The requirement is to convince people of the dangers of heart disease. The theme is a persuasive argument that first tries to convince people of the dangers from smoking, poor diet, stressed lifestyles, and so on, then explains how to improve their lifestyle to prevent heart disease, followed by reinforcing the message with the benefits of a healthy lifestyle such as lower health insurance, saving money, longer life. Subthemes are embedded at different points so users can explore the facts behind heart disease, the statistics and their exposure, and how to get help. Information is then gathered for each node in the thematic map. How this architecture will be delivered depends on interaction design decisions: it could become an interactive story to explore different lifestyle choices, combined with a quiz. The outcome of information architecture analysis will be an information-enhanced task model, a thematic map, or a hierarchy/network to show the structure and relationships of information categories. The next step is to analyze the information content by classifying it by types.

Information types are amodal, conceptual descriptions of information components that elaborate the content definition. Information components are classified into one or more of the following:

- Physical items relating to tangible observable aspects of the world
- Spatial items relating to geography and location in the world
- Conceptual-abstract information, facts, and concepts related to language
- Static information which does not change: objects, entities, relationships, states, and attributes
- Dynamic, or time-varying information: events, actions, activities, procedures, and movements
- Descriptive information, attributes of objects and entities
- Values and numbers
- Causal explanations

More complex taxonomies elaborate concepts and linguistic information as ontologies and arguments (Mann and Thompson 1988), but additional complexity is only warranted for tools that automatically generate multimedia output (Zhou and Feiner 1998). It is important to note that one component may be classified with more than one type; for instance, instructions on how to get to the railway station may contain procedural information (the instructions <turn left, straight ahead, etc.>) and spatial or descriptive information (the station is in the corner of the square, painted blue). The information types are "tools for thought" that can be used either to classify specifications of content or to consider what content may be necessary. To illustrate, for the task "navigate to the railway station," the content may be minimally specified as "instructions how to get there," in which case the information types prompt questions in the form "what sort of information does the user need to fulfill the task/user goal?" Alternatively, the content may be specified as a scenario narrative of directions, waymarks to recognize, and description of the target. In this case, the types classify components in the narrative to elucidate the deeper structure of the content. The granularity of components is a matter of the designer's choice and will depend on

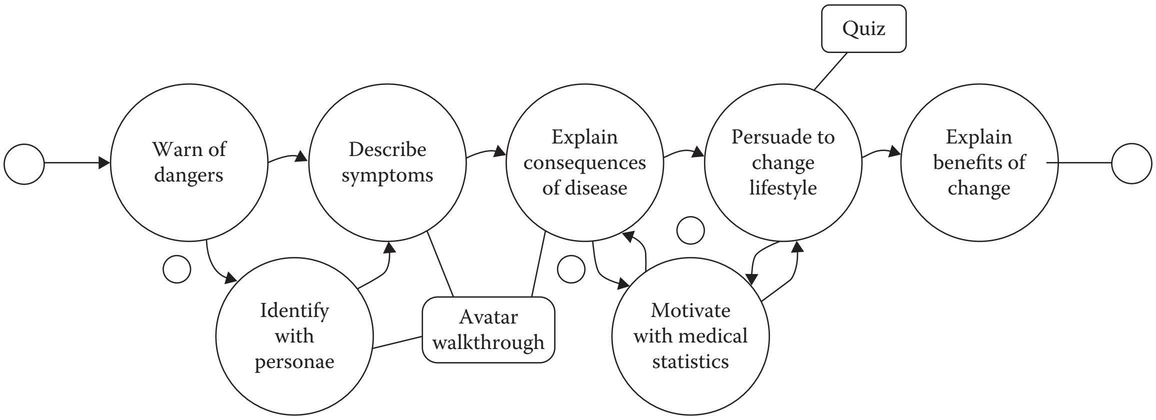

Quiz
Warn of dangers
Describe symptoms
Explain consequences of disease
Persuade to change lifestyle
Explain benefits of change
Identify with personae
Avatar walkthrough
Motivate with medical statistics

FIGURE 17.3 Thematic map for a healthcare promotion application.

the level of detail demanded by the application. To illustrate the analysis:

Communication goal: Explain how to assemble a bookshelf from ready-made parts.

Information component 1:
  Parts of the bookshelf, sides, back, shelves, connecting screws
  Mapping to information types:
    Physical-static-descriptive; parts of the bookshelf are tangible, do not change and need to be described
    Physical-static-spatial; dimensions of the parts, how they are organized
    Physical-static-relationship type could also be added to describe which parts fit together
Information component 2:
  How to assemble parts instructions
  Mapping to information types:
    Physical-dynamic-discrete action
    Physical-dynamic-procedure
    Physical-static-state; to show final assembled bookshelf

17.4.3 Media Selection and Combination

The information types are used to select appropriate categories of media resource(s). Media classifications focus on the psychological properties of the representations rather than the physical nature of the medium (e.g., digital or analogue encoding in video). Note that these definitions are combined to describe any specific medium, so speech is classified as an audio, linguistic medium, whereas a cartoon is classified as a nonrealistic (designed) moving image.

The definitions may be usefully considered in two dimensions of abstraction: the designer's involvement in creating the medium and rate of change. Media resources are classified using the decision tree illustrated in Figure 17.4. More fine-grained taxonomic distinctions can be made, for instance, between different signs and symbolic languages (see Bernsen [1994]), but as with information types, richer taxonomies increase specification effort.

The approach to classifying media uses a walk-through of the decision tree with the following questions that reflect the facets of the classification:

- Is the medium perceived to be realistic or not? Media resources captured directly from the real world will usually be realistic, for example, photographs of landscapes, sound recordings of bird song. In contrast, nonrealistic media are created by human action. However, the boundary case category that illustrates the dimension is a realistic painting of a landscape.
- Does the medium change over time or not? The boundary case here is the rate of change, particularly in animations where some people might judge 10 frames/s to be a video, but 5 slides in 1 minute shown by a PowerPoint presentation to be a sequence of static images.
- Which modality does the resource belong to? In this case, the categories are orthogonal, although one resource may exhibit two modalities; for example, a film with a soundtrack communicates in both visual and audio modalities.

Classification of media resources facilitates mapping of information types to media resources; however, the process may also guide the acquisition or creation of new resources,

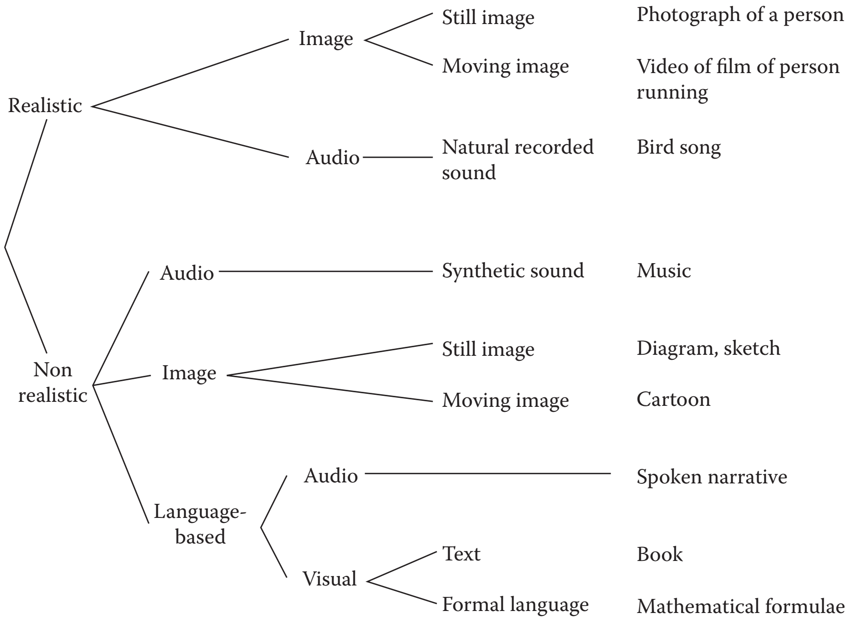


FIGURE 17.4 Decision tree for classifying media resources.

if appropriate resources are not present in the designer's media resource library. Finally, the classification provides a mechanism for indexing media resource libraries.

17.4.3.1 Media Selection

Recommendations for selecting media have to be interpreted according to the users' task and design goal. If information provision is the main design goal—for example, a tourist kiosk information system—then persistence of information and drawing attention to specific items is not necessarily as critical as in tutorial applications. Task and user characteristics influence media choice; for instance, verbal media are more appropriate to language-based and logical reasoning tasks; visual media are suitable for spatial tasks involving moving, positioning, and orienting objects. Some users may prefer visual media, whereas image is of little use for blind users. Media resources may be available for selection, or have to be purchased from elsewhere. If existing media can be edited and reused, this is usually preferable to creating new media from scratch. Graphical images can be particularly expensive to draw, whereas capture of images by scanning is usually quick and cheap. The following heuristics are supplemented by more detailed examples in Table 17.1.

- To convey detail, use static media, for example, text for language-based content, diagrams for models, or still image for physical detail of objects (Booher 1975; Faraday and Sutcliffe 1998).
- To engage the user and draw attention, use dynamic media—video for physical information, animation, or speech, for example.
- For spatial information, use diagrams, maps, with photographic images to illustrate detail, animations to indicate pathways (Bieger and Glock 1984; May and Barnard 1995).
- For values and quantitative information, use charts and graphs for overviews and trends, supplemented by tables for detail (Bertin 1983; Tufte 1997).
- Abstract concepts, relationships, and models should be illustrated with diagrams explained by text captions and speech to give supplementary information.
- Complex actions and procedures should be illustrated as a slideshow of images for each step, followed by a video of the whole sequence to integrate the steps. Text captions on the still images and speech commentary provide supplementary information (Hegarty and Just 1993). Text and bullet points summarize steps at the end, so choice trade-offs may be constrained by cost and quality considerations.
- To explain causality, still and moving image media need to be combined with text (Narayanan and Hegarty 1998). For example, the cause of a flood is explained by text describing excessive rainfall with an animation of the river level rising and overflowing its banks. Causal explanations of physical phenomena may be given by introducing the topic using linguistic media, showing the cause and effect by a combination of still image and text with speech captions for commentary; integrate the message by moving image with voice commentary and provide a bullet point text summary.

Because most components in the information architecture will have multiple information types and each information type may match several media, the selection process encourages multimedia integration. For example, when a procedure for explaining a physical task is required, first a series of realistic images will be selected, followed by video, and speech to integrate the steps, then text to summarize the key points.

The end point of media selection is media integration: one or more media will be selected for each information group to present complementary aspects of the topic. Some examples of media combination that amplify the basic selection guidelines are given in Table 17.1.

17.5 MEDIA DESIGN FOR USER ENGAGEMENT

The design process in Section 17.4 was oriented to a task-driven view of media. However, multimedia design is frequently motivated by the need to attract users' attention and to make the user experience interesting and engaging. These considerations may contradict some of the earlier guidelines because the design objective is to please the user and capture their attention rather than deliver information effectively. First, a health warning should be noted: The old saying "beauty is in the eye of the beholder" has good foundation. Judgments of aesthetic quality suffer from considerable individual differences. A person's reaction to a design is a function of their motivation (see Chapter 4 from the Second Edition of the HCI Handbook). Individual preferences, knowledge of the domain, and exposure to similar examples, to say nothing of peer opinion and "fashion." Furthermore, attractiveness is often influenced more by content than the choice of media or presentation format. The following guidelines should, therefore, be interpreted with care and their design manifestations tested with users.

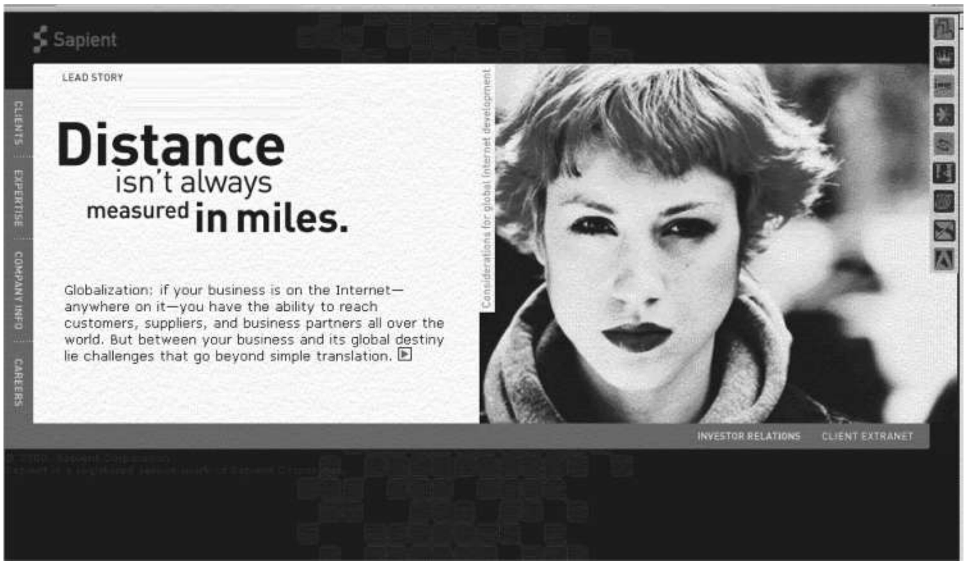
17.5.1 Multimedia to Motivate and Persuade

Design of media for motivation is a complex area in its own right, and this topic is dealt with in more depth in Chapter 7 of the Second Edition (Fogg), so the treatment here will focus on media selection issues. Simple photographs or more complex interactive animations (talking heads or full body mannequins) have an attractive effect. We appear to ascribe human properties to computers when interfaces give human-like visual cues (Reeves and Nass 1996); however, the effectiveness of media representing people depends on the characters' appearance and voice; see Figure 17.5. In human–human conversation, we modify our reactions according to our knowledge, or assumptions about, the other person's role, group identification, culture, and intention (Clark 1996). For example, reactions to a military mannequin will be very different from those to the representation of a parson. Male voices tend to be treated as more authoritative than female voices.

TABLE 17.1
Media Selection Example

Information Type Media Type	Causation	Conceptual	Continuous Action	Descriptive	Discrete Action	Event	Physical	Procedure	Relationship	Spatial Information	State	Value
Realistic audio	Sound of rain and storms		Sound of skiing		Click of ON switch	Sound of the starting gun	*Noise of a tornado*			Echoes in a cave	Sound of snoring	Musical note encodes a value
Nonrealistic audio		Rising tone illustrates increasing magnetic force	Continuous tone signals progress of action	Morse code describes a ship	Tones signal open/close door	*Alarm siren*			Tones associate two objects	Sonar and Doppler effect	Continuous sound in a heartbeat monitor	
Speech	*Tell someone why El Nino happens*	Tell someone about your religious beliefs	Tell someone what a ski turn looks like	Verbal description of a person	Tell someone how to turn computer on	*Tell someone race has started*	Tell someone how it feels to be in a storm	Speak instructions on engine assembly	Tell someone Jack and Jill are related	Tell someone pathway to and location of railway station	Tell someone "Jane's asleep"	Verbal report of numbers, figures
Realistic still image	*Photograph of El Nino storms and ocean currents*	Statue of Liberty photograph represents "freedom"	Set of photographs showing snap shots of action	*Overview and detail photographs of a car*	*Photograph of computer ON switch*	Photograph of the start of a race	*Photograph of a person's face*	*Photographs showing engine assembly*		*Photograph of a landscape*	Photograph of a person sleeping	
Nonrealistic still image	*Diagrams of ocean currents and sea temp. to explain El Nino*	*Hierarchy diagram of plant taxonomy*	Diagram with arrow depicting ski turn motion	Histogram of aging population	Diagram showing where and how to press ON switch	Event symbol in a race sequence diagram		Explode parts diagram of engine with assembly numbers	*Graphs, histograms,* ER *diagrams*	*Map of the landscape*	Waiting state symbol in race sequence diagram	*Charts, graphs, scatter plots*
Text	*Describe reasons for El Nino storms*	*Explain taxonomy of animals*	Describe ski turn action	*Describe a person's appearance*	Describe how to turn computer on	Report that the race has started	Report of the storm's properties	*Bullet point steps in assembling engine*	*Describe brother and sister relationship*	Describe dimensions of a room	*Report that the person is asleep*	*Written number one, two*
Realistic moving image	*Video of El Nino storms and ocean currents*		*Movie of person turning while skiing*	Aircraft flying		*Movie of the start of a race*	*Movie of a storm*	*Video of engine assembly sequence*		Fly through landscape	Video of a person sleeping	
Nonrealistic moving image	*Animation of ocean temperature change and current reversal*	Animated diagram of force of gravity	Animated mannequin doing ski turn		Animation showing operation of ON switch	Animation of start event symbol in diagram		*Animation of parts diagram in assembly sequence*	Animation of links on ER diagram			
Language-based: formal, numeric	Equations, functions formalizing cause and effect	*Symbols denoting concepts, for example, pi*			Finite state automata	Event-based notations		Procedural logics, process algebras	Functions, equations, grammars		State-based languages, for example, Z	numeric symbols

The table summarizes the media selection and combinations for each information type. The italics denote the preferred mappings for media and information types, whereas ordinary text shows other potential media uses for the information type.



Distance isn't always measured in miles.

Globalization: if your business is on the Internet—anywhere on it—you have the ability to reach customers, suppliers, and business partners all over the world. But between your business and its global destiny lie challenges that go beyond simple translation. ▶

FIGURE 17.5 Effective use of human image for attraction. The picture attracted by the direction of gaze to the user as well as by the appearance of the individual.

Use of human-like forms is feasible with prerecorded video and photographs; however, the need depends on the application. Video representation of the lecturer can augment presentations, and video communication helps interactive dialogue. A good speaker holds our attention by a variety of tricks, such as maintaining eye contact, varying the voice tone, using simple and concise language, and delivering an interesting message. These general effects can be reinforced by projected personality. Friendly people are preferred over colder, more hostile individuals. TV announcers who tend to be middle-aged, confident, but avuncular characters have the attention-drawing power of a dominant yet friendly personality. Both sexes pay attention to extrovert, young personalities, whereas the male preference for beautiful young women is a particularly strong effect. These traits have been exploited by advertisers for a long time. There are lessons here for multimedia designers as the web and interactive TV converge, and when we want media to convey a persuasive message (Reeves and Nass 1996; Fogg, this book). Media selection guidelines for motivation and persuasion, adapted from Reeves and Nass (1996), can be summarized as follows:

- Human image and speech invokes the computer-as-social actor effect to facilitate motivation and persuasion.
- Photographs of people attract attention especially when the person is looking directly at the user.
- Faces that represent the norm in a population (Mr./Ms. average) and young children are more attractive. We are very susceptible to the large-eyes effect in young animals, as exploited by Disney cartoons.
- Polite praise: Use of "Please," "Thank you," and simple compliments like "That was an excellent choice" increase people's tendency to judge the computer as pleasant and enjoyable.
- Short compelling argument: Such as the well-known British World War I recruiting poster featuring General Kitchener gazing directly at the viewer with the caption "Your country needs you."

For more detailed treatment of design for persuasive technology, see Fogg, Chapter 7, Second Edition of the HCI Handbook.

17.5.2 Media for Emotional Effects

Media design for affect (emotional response and arousal) involves both choice of content and interaction. Arousal is increased by interactive applications, surprising events during interaction, use of dynamic media, and challenging images. In contrast, if the objective is to calm the users, arousal can be decreased by choice of natural images and sounds, and soothing music. The most common emotional responses that designers may want to invoke are pleasure, anxiety and fear, and surprise. Pleasure, anxiety, and fear usually depend on our memory of agents, objects, and events (Ortony, Clore, and Collins 1988), so content selection is the important determinant. Anxiety can be evoked by uncertainty in interaction and cues to hidden effects, whereas emotional response of fear or pleasure will depend on matching content to the user's previous experience. Some guidelines to consider are as follows:

- *Dynamic media*, especially video, have an arousing effect and attract attention; hence, video and animation are useful in improving the attractiveness of presentations. However, animation must be used with care, as gratuitous video which cannot be turned off quickly offends (Spool et al. 1999).

- *Speech* engages attention because we naturally listen to conversation. Choice of voice depends on the application: female voices for more restful and information effects, male voices to suggest authority and respect (Reeves and Nass 1996).
- *Image*s may be selected for mood setting, for example, to provide a restful setting for more important foreground information (Mullet and Sano 1995). Backgrounds in half shades and low-saturation color provide more depth and interest in an image.
- *Music* has an important emotive appeal, but it needs to be used with care. Classical music may be counterproductive for a younger audience, whereas older listeners will not find heavy metal pop attractive. Music can set the appropriate mood, for example, loud strident pieces will arouse and excite, romantic music calms and invokes pleasure, and so on.
- *Natural sounds* such as running water, wind in trees, bird song, and waves on a sea shore have restful properties and hence decrease arousal.
- *Dangerous and threatening episodes*, for example, being chased by a tiger, gory images (mutilated body), and erotic content all increase arousal and invoke emotions ranging from fear to anger, whereas pleasant images (e.g., flowers, sunset) tend to decrease it, that is, have calming effects and produce pleasurable emotional responses.
- *Characters* can appear threatening or benevolent depending on their appearance or dress. For example, disfigured people appear threatening and evoke emotions ranging from fear to disgust. Characters familiar from popular culture can be used for emotional effect.
- *Dialogue* is probably the most powerful tool for creating emotional responses, from threats to empathy. Emotional effects are additive so choice of character with a threatening appearance, complemented by a menacing voice tone and an aggressive dialogue, all reinforce the emotions of anxiety and fear.

Media integration rules may be broken for emotive effects. For example, use of two concurrent video streams might be arousing for a younger audience, as music TV (MTV) and pop videos indicate. Multiple audio and speech tracks can give the impression of complex, busy, and interesting environments.

17.5.3 Multimedia and Aesthetic Design

If the requirements analysis indicates that having a pleasing and attractive design is important for the user's perception, then aesthetics need to be considered in depth. However, aesthetics should be considered as a design criterion for all applications since poor appearance and interaction design may provoke adverse reaction (Norman 2004). Some studies suggest that aesthetic design is an important component of usability and overall preference (Tractinsky 1997; Tractinsky, Shoval-Katz, and Ikar 2000; Hassenzahl 2004); however, others have shown that aesthetic preferences are open to contextual effects on users' judgment (Sutcliffe and De Angeli 2005; Hartmann, Sutcliffe, and De Angeli 2008). Judging when aesthetics may be important is not easy. For example, in e-commerce applications with high-value, designer-label products, aesthetic presentation is advisable; similarly, when selling to a design-oriented audience. However, in many applications, the decision is not clear-cut.

Aesthetic design primarily concerns graphics and visual media. Evaluation questionnaires assess design on classic aesthetics, which broadly equate with conventional usability guidelines on structured and consistent layout, and expressive aesthetics that capture the more creative aspects of visual design (Lavie and Tractinsky 2004); however, these measure user reaction to general design aspects such as "original," "fascinating," "clear," and "pleasant." The following heuristics provide more design-directed guidance, but they may also be used for evaluation (Sutcliffe 2002; Sutcliffe and De Angeli 2005).

- *Judicious use of color*: Color use should be balanced and low-saturation pastel colors should be used for backgrounds. Designs should not use more than two to three fully saturated intense colors. Yellow is salient for alerting, red/green have danger/safety positive/negative associations, and blue is more effective for background. Low-saturated colors (pale shades with white) have a calming effect and are also useful for backgrounds. Color is a complex subject in its own right; for more guidance, see Travis (1991).
- *Depth of field*: Use of layers in an image stimulates interest and can attract by promoting curiosity. Use of background image with low-saturated color provides depth for foreground components. Use of layers in an image and washed-out background images stimulate curiosity and can be attractive by promoting a peaceful effect.
- *Use of shape*: Use of curved shapes conveys an attractive visual style, in contrast to blocks and rectangles which portray structure, categories, and order in a layout.
- *Symmetry*: Symmetrical layouts, for example, bilateral, radial organization that can be folded over to show the symmetrical match.
- *Simplicity and space*: Uncluttered, simple layout that uses space to separate and emphasize key components.
- *Design of unusual or challenging images* that stimulate the users' imagination and increase attraction: Unusual images often disobey normal laws of form and perspective.
- *Visual structure and organization*: Dividing an image into thirds (right, center, left; or top, middle, bottom) provides an attractive visual organization,

whereas rectangular shapes following the golden ratio (height/width = 1.618) are aesthetically pleasing. Use of grids to structure image components promotes consistency between pages.

Although guidelines provide ideas that can improve aesthetic design and the attractiveness of interfaces, they are no guarantee that these effects will be achieved. Design is often a trade-off between ease of use and aesthetic design; for instance, use of progressive disclosure to promote flow may well be perceived by others as being difficult to learn. Visual effects often show considerable individual differences and learning effects, so a well-intentioned design might not be successful. The advice, as with most design, is test ideas and preliminary designs with users to check interpretations, critique ideas, and evaluate their acceptability. There are several sources of more detailed advice on aesthetics and visual design (Kristoff and Satran 1995, Mullet and Sano 1995; Lidwell, Holden, and Butler 2003); however, advice is usually given as examples of good design rather than specific guidelines.

17.6 INTERACTION AND NAVIGATION

Although discussion of interactive multimedia has been delayed until now, in practice, dialogue and presentation design proceed hand in hand. Task analysis provides the basis for dialogue design and specification of navigation controls. Navigational and control dialogues allow flexible access to the multimedia content and enable users to control how media are played. Dialogue design may also involve specifying how users interact with tools, agents, and objects in interactive microworlds.

17.6.1 Metaphors and Interaction Design

Although task and domain analysis can provide ideas for interaction design, this is also a creative process. Interaction design is essentially a set of choices along a dimension from simple controls such as menus and buttons where the user is aware of the interface, to embodiment in which the user becomes involved as part of the action by controlling an avatar or other representation of their presence. At this end of the dimension, multimedia interaction converges with VR (see Chapter 29). Interactive metaphors occupy the middle ground.

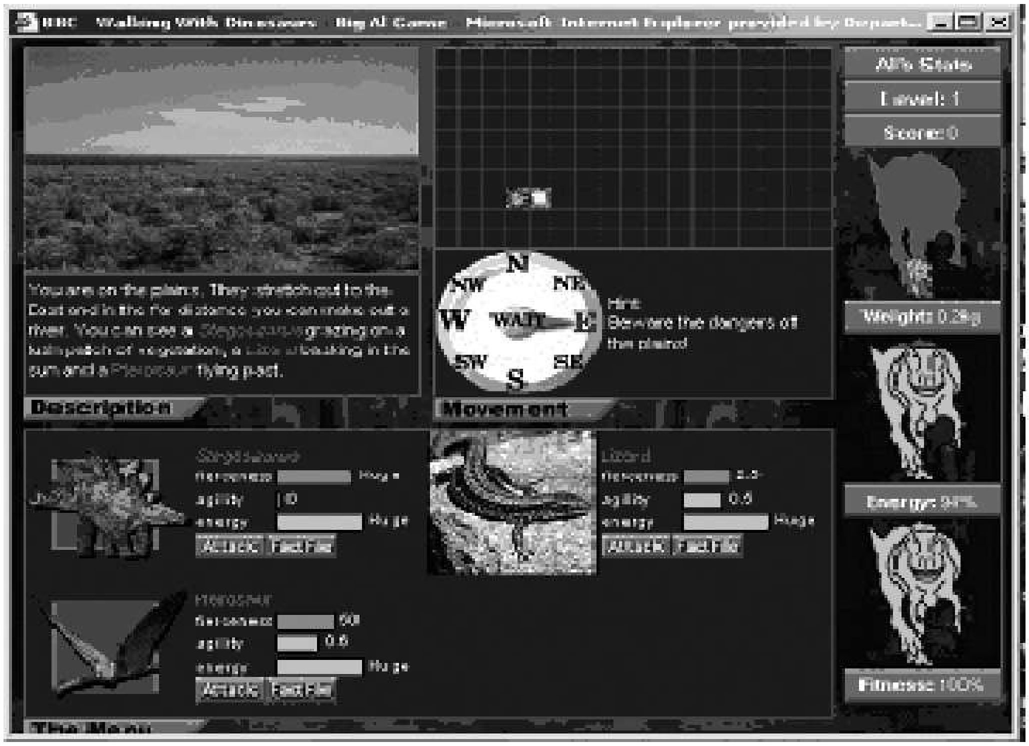
Some interactive metaphors are generally applicable, such as timelines to move through historical information, the use of a compass to control direction of movement in an interactive space, controls based on automobiles (steering wheels) or ships (rudders). Others will be more specific, for example, selecting and interacting with different characters (young, old, male, female, overweight, fit, etc.) in a health-promotion application. Design of interaction also involves creating the microworld within which the user moves and interactive objects that can be selected and manipulated.

Interaction via characters and avatars can increase the user's sense of engagement first by selecting or even constructing the character, although some users may not have the patience to build their own avatar using a graphical paint program. In character-based interaction, the user can either see the world from an egocentric viewpoint, that is, from their character's position, or exocentric when they see their character in the graphical world. The sophistication in control of movement and interaction will depend on the hardware available (e.g., joystick, wand, or standard mouse and keyboard). Although mimicking physical interaction via data gloves and tracking requires VR technology, relatively complex interaction (e.g., actions in a football game, pass, head ball in direction north/south/east/west) can be programmed using buttons and function keys. Engagement is also promoted by surprise and unexpected effects, so as the user moves into a particular area, a new subworld opens up, or system-controlled avatars appear. These techniques are well known to games programmers; however, they are also applicable to other genres of multimedia applications. The design concepts for engagement can be summarized as follows (see Chapter 32):

- *Character-driven interaction*: This interaction provides the user with a choice of avatars or personae they can adopt as representations of themselves within the interactive virtual world; see Figure 17.6. Avatar development tools enable virtual characters to be designed and scripted with actions, and simple speech dialogues. Most sophisticated semi-intelligent "chatterbots" (e.g., Alice, Jabberwocky*) use response-planning rules to analyze user input and generate naturally sounding output; however, it is easy to fool these systems with complex natural language input.
- *Tool-based interaction*: This places tools in the world which users can pick up; the tool becomes the interface, for example, a virtual mirror magnifies, a virtual helicopter flies (Tan, Robertson, and Czerwinski 2001).
- *Collaborative characters*: In computer-mediated communication, these characters may represent other users; in other applications, system-controlled avatars appear to explain, guide, or warn the user.
- *Surprise effects*: Although conventional human–computer interaction (HCI) guidelines should encourage making the affordances and presence of interactive objects explicit, when designing for engagement, hiding, and surprise are important.

Interaction design for an explanatory/tutorial application is illustrated in Figure 17.6. This is an interactive microworld in which the user plays the role of a dinosaur character, illustrating use of the engagement concepts. A compass navigation metaphor allows the user to act as the dinosaur moving

*http://alice.pandorabots.com/ and http://www.jabberwacky.com


You are on the plains. They stretch out to the East and in the far distance you can make out a river. You can see a Stegosaurus grazing on a lush patch of vegetation, a Lizard basking in the sun and a Pterosaur flying past.

Hint:
Beware the dangers of the plains!

FIGURE 17.6 Interactive microworld: Big Al game (www.bbc.co.uk/sn/). The user plays the dinosaur role by navigating with the compass metaphor. The photograph updates with each move and the user is given a choice of attacking or avoiding other dinosaurs in the virtual world.

around the landscape illustrated in photographs. The user is given feedback on the characteristics of other predators and prey in the vicinity and has to decide whether to attack or avoid them. Other controls that might be added to such interactive microworlds could be settings to change the environment, for example, add more predators, change the weather, and so on. Engagement can be taken even further by giving the user facilities to actually design the MacWorld so the application becomes a domain-oriented design environment (Fischer et al. 2004).

17.6.2 Navigation

In information-intensive multimedia where access to content is the main design goal, hypermedia dialogues that link content segments will be appropriate. Good hypertext design is based on a sound information analysis that specifies the pathways between related items, and use of cues to show the structure of the information space to the user. In document-based hypermedia (e.g., HTML and the web), links can only access the whole media resource rather than point to components within it. The access structure of most hypermedia will be hierarchical, organized according to the information model and categorization of content, for example, information grouped by function, organization, task usage, or user preference. Navigation design transforms the user's conceptual model of an information space into a hypermedia structure. Unfortunately, individual users have different models so this may not be an easy task. Implementing too many links to satisfy each user's view will make the system too complex and increase the chance of the user getting lost. Too few links will frustrate users who cannot find the associations they want. Unfortunately, hypermedia systems assume a fixed link structure so the user is limited to the pathways provided by the designer. More open-ended hypermedia environments (e.g., Microcosm: Lowe and Hall 1998) provide more flexibility via links with query facilities which allow access to databases. Dynamic links attached to hotspots in images or nodes in text documents provide access paths to a wider variety of data.

One problem with large hypermedia systems is that users get lost in them. Navigation cues, waymarks, and mini-map overviews can help to counter the effects of disorientation. *Mini-maps* give an overview of the hypertext area and a reference context for where users are in large networks. *Filters* help to reduce complexity by showing only a subset of nodes and links that the user is interested in. Having typed links helps filtering views because the user can guess the information content from the link type, for example, reference, example, source, related work, and so on. Other navigation facilities are *visit lists* containing a history of nodes traversed in a session and *bookmarks* so users can tailor a hypermedia application with their own navigation aide-memoires (Nielsen 1995). Once the structure has been designed, access structures and link cues need to be located within media resources, so the appropriate cues need to be considered for each medium, such as the following, for example.

- *Text media*: The web convention is to underline and highlight text in a consistent color, for example, blue or purple. Text image thumbnails can be used to illustrate document and page structure, to facilitate direct pointing access, as used in the Adobe PDF Reader.
- *Images*: Link cues can be set as standalone icons or as active components in images. Icons need to be tested with users because the designer's assumed meaning can be ambiguous. Active components should signal the link's presence by captions or pop-up hover text so the user can inspect a link before deciding whether to follow it. Mosaics of image thumbnails are an alternative access path and can be organized in dimensions or layers to communicate categories and properties, such as time × location dimensions, general views to close-ups arranged in concentric circles of magnification. Slideshow presentation of images organized in navigation sequences with a stop button is another option for rapid access (De Bruijn, Spence, and Chong 2002).
- *Moving images*: Links from animation and film are difficult to design because the medium is dynamic; however, link buttons can be placed below the video window. Active components, for example, overlaid buttons within a moving image, are technically more challenging to program. Buttons may also be timed to pop-up at appropriate times during the video. Sample frames set in a mosaic and timeline structure summarize videos and enables access by pointing to segments. This can be taken further with multirunning movie thumbnails to provide overviews; however, this can produce more distraction than useful navigation, and many instances of dynamically running media overload our attention (see Section 17.3).
- *Sound and speech* links are difficult for the same reason as with moving images. One solution is to use visual cues, possibly synchronized with the sound or speech track. If speech recognition is available, then voice commands can act as links, but these commands need to be explained to the user. Visual access structures can be based on sonograms of the audio track or more usefully annotated timelines.

In many cases, controls will be provided by the media-rendering device, for example, video player for .avi files, or Quicktime movies. If controls have to be implemented from scratch, the following should be considered for each media type:

- *Static media*. Size and scale controls to zoom and pan; page access if the medium has page segmentation, as in text and diagrams; the ability to change attributes such as color, display resolution, font type, and size in text.
- *Dynamic media*. The familiar video controls of stop, start, play, pause, fast-forward, and rewind, also the ability to address a particular point or event in the media stream by a time marker or an index, for example, "go to" component/marker, and so on.

Navigation controls use standard UI components (buttons, dialogue boxes, menus, icons, sliders) and techniques (form filling, dialogue boxes, and selection menus); for more guidance, see ISO 9241, Parts 12, 14, and 17 (ISO 1997) and ISO 14915 Part 2 (ISO 1998).

17.6.3 Design for Attention

Having selected the media resources, the designer must now ensure that the user will extract the appropriate information. An important consideration of multimedia design is to link the thread of a message across several different media. This section gives recommendations on planning the user's reading/viewing sequence, and guidelines for realizing these recommendations in presentation sequences, hypermedia dialogues, and navigation controls. The essential differences are timing and user control. In a presentation design, the reading/viewing sequence and timing are set by the designer; whereas the reading/viewing sequence in hypertext implementation and interactive dialogues is under user control.

Presentation techniques help to direct the user's attention to important information and specify the desired order of reading or viewing. Thematic links between information components are specified and attention-directing techniques are selected to implement the desired effect.

The design issues are as follows:

- To plan the overall thematic thread of the message
- To draw the user's attention to important information
- To establish a clear reading/viewing sequence
- To provide clear links when the theme crosses from one medium to another

Design for attention is particularly important for images. User attention to time-varying media is determined by the medium itself, that is, we have little choice but to listen to speech or to view animations in the order in which they are presented. The reading sequence is directed by the layout of text, although this is culturally dependent, for example, western languages read left to right, Arabic in the opposite direction. However, viewing order in images is unpredictable unless the design specifically selects the user's attention.

The design problem is how to direct the user's attention to the appropriate information at the correct level of detail. Initially, users will tend to extract information from images at the scene level, that is, major objects will be identified but with very little descriptive detail (Treisman 1988). Regular layout grids help design composite images (Mullet and Sano 1995) and encourage viewing sequences in image sets. Alternatively, the window frame can be set to control which parts of an image are viewed. Larger window frames will be attended to before smaller areas. A list of the key components that the user needs to focus on and the facts that should be extracted are checked against the initial presentation design to see if the key components will attract sufficient attention or whether the user is likely to be confused by extraneous detail.

Position on screen is a key influence on attention. Eye-tracking studies have demonstrated that components on the top-left and center areas of screens receive more attention than the lower and right-hand side (Beymer, Orton, and Russell 2007; Hornof and Halverson 2003). Furthermore, large centralized images tend to dominate attention while layout structure, such as columns and blocks focus users' gaze within these structures. So layout, as well as media choice, can be used to influence attention; then within each media type, attention can be directed by applying the following highlighting techniques.

17.6.3.1 Still Image Media

Highlighting techniques for designed and natural images, organized in approximate power of their effect, are summarized in Table 17.2. A common highlighting technique will pick out spatially distributed objects, for example, change all the related objects to the same color; co-located objects can be grouped by using a common color or texture for their background or drawing a box around them. The highlighted area will set the granularity of the user's attention. Captions linked to objects in an image are another useful means of drawing attention and providing supplementary information (e.g., identity). Dynamic revealing of captions is particularly effective for directing the user's viewing sequence. Sequential highlighting is also useful for showing the pathways or navigational instructions.

17.6.3.2 Moving Image Media

Directing attention to components within moving images is difficult because of the dynamic nature of the medium. Design of film and video is an extensive subject in its own right, so treatment here will necessarily be brief. The following design advice is based on Hochberg (1986). The design objectives, as for other media, are how to draw the user's attention to key components within the video or animation.

First, the content needs to be structured into scenes that correspond to the information script. To structure animation sequences and make scene boundaries obvious, use a cut, wipe, or dissolve to emphasize that a change in the content structure has taken place. However, cuts should be used with care and continuity maintained between the two sequences if they are to be integrated. Continuity is manifest as the same viewpoint and subject matter in two contiguous shots. Change in background or action, such as an individual walking left in one clip and walking right in the next, is quickly noticed as a change. An establishing shot that shows the whole scene helps to introduce a new sequence and provide context. To provide detail of a newly introduced object or context, the object is shown filling the frame with a small amount of surrounding scene; while to imply a relationship or compare two objects a tight two shot with both objects together in the same frame is advisable.

17.6.3.3 Linguistic Media (Text and Speech)

As with moving image, the literature is extensive, so the following heuristics are a brief summary; see Levie and Lentz (1982) for more detail. Text may be structured to indicate subsections by indentation, formatting into paragraphs, columns, or segmented by background color. Bullet points or

TABLE 17.2
Attention-Directing Techniques for Different Media

	Attention-Orienting Techniques in Approximate Order of Power	Notes
Still image: designed and natural	Movement of or change in the shape/size/color of an object. Use of bold outline. Object marked with a symbol (e.g., arrow) or icon. Draw boundary, use color, shape, size, or texture to distinguish important objects.	Some effects may compromise natural images because they overlay the background image with new components (e.g., arrows, arcs, icons). Group objects by a common technique.
Moving image	Freeze frame followed by applying a still image highlight. Zoom, close-up shot of the object. Cuts, wipes, and dissolve effects.	Change in topographic motion, in which an object moves across the ground of an image, is more effective than internal movement of an object's components. Size and shape may be less effective for highlighting a moving object.
Text	Bold, font size, type, color, or underlining. To direct attention to larger segments of text, use formatting, bullet points, sub-sections, indentation.	Formatting techniques are paragraphs; headings/titles as entry points; indents to show hierarchical nesting, with bullet points and lists.
Speech/sound	Familiar voice. Silence followed by onset of sound. Different voices, or a change in voice prosody (tonality), amplitude (loudness), change, and variations in pitch (frequency), voice rate, change source direction, alarm sounds (police sirens).	Voices familiar to the user (e.g., close relatives) attract attention over nonfamiliar speech. Discourse markers "next," "because," "so," and so on draw attention to subsequent phrases.

numbered sections indicate order more formally, such as for procedures. Different voices help to structure speech while also attracting attention. If language is being used to set the context for accompanying media, it is important that the correct level of identification is set. For instance, a higher level concept, or the whole scene in an accompanying image, is described at the beginning of a script, and then lower level topics reset the user's focus. Discourse markers can make phrases and sentences more salient.

Adding attention-directing effects completes the design process; however, as with all UIs, there is no substitute for usability testing. Designs are constructed incrementally by iterations of design and evaluation that checks for usability using standard methods, with additional memory and comprehension tests for multimedia. So when testing a design, ask the user to tell you what they understood the message to be. This can be done during the presentation with a think-aloud protocol to check that users did attend to key items, and afterwards by a memory test. If key components in the message are not being remembered, then the design may need to be improved.

17.7 CONCLUSIONS

Multimedia still poses many issues for further research. The design method described in this chapter coupled with user-centered design can improve quality; however, there is still a need for experts to create specific media resources, for example, film/video, audio experts. Furthermore, considerable research is still necessary before we fully understand the psychology of multimedia interaction. Design for motivation and attractiveness is still poorly understood, and personality effects in media may not be robust when usability errors impede communication. The process by which people extract information from complex images still requires extensive research, although the increasing number of eye-tracking studies is beginning to throw some light on this topic. In the future, language and multimodal communication will change our conception of multimedia from its current CD-ROM or web-based form into interfaces that are conversational and multisensory. Multimedia will become part of wearable and ubiquitous UIs where the media is part of our everyday environment. Design for multisensory communication will treat media and artifacts (e.g., our desks, clothes, walls in our homes) as a continuum, whereas managing the diverse inputs to multimedia from creative design, technology, and usability engineering will be one of the many interesting future challenges.

REFERENCES

Alty, J. L. 1991. Multimedia: What is it and how do we exploit it? In *Proceedings of HCI '91: People and Computers VI*, ed. D. Diaper and N. V. Hammond, 31–41. Cambridge: Cambridge University Press.

Baddeley, A. D. 1986. *Working Memory*. Oxford: Oxford University Press.

Barnard, P. 1985. Interacting cognitive subsystems: A psycholinguistic approach to short term memory. In *Progress in Psychology of Language*, ed. A. Ellis, Vol. 2, 197–258. London: LEA.

Bernsen, N. O. 1994. Foundations of multimodal representations: A taxonomy of representational modalities. *Interact Comput* 6(4):347–71.

Bertin, J. 1983. *Semiology of Graphics*. Madison, WI: University of Wisconsin Press.

Beymer, D., P. Z. Orton, and D. M. Russell. 2007. An eye tracking study of how pictures influence online reading. In *Proceedings of Interact 2007*, 456–60. Berlin, Germany: Springer.

Bieger, G. R., and M. D. Glock. 1984. The information content of picture-text instructions. *J Exp Educ* 53:68–76.

Booher, H. R. 1975. Relative comprehensibility of pictorial information and printed word in proceduralized instructions. *Hum Factors* 17(3):266–77.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Clark, H. H. 1996. *Using Language*. Cambridge: Cambridge University Press.

De Bruijn, O., R. Spence, and M. Y. Chong. 2002. RSVP Browser: Web browsing on small screen devices. *Pers Ubiquitous Comput* 6(4):245–52.

Faraday, P., and A. G. Sutcliffe. 1996. An empirical study of attending and comprehending multimedia presentations. In *Proceedings ACM Multimedia '96: 4th Multimedia Conference, Boston, MA 18–22 November 1996*, 265–75. New York: ACM Press.

Faraday, P., and A. G. Sutcliffe. 1997. Multimedia: Design for the moment. In *Proceedings: Fifth ACM International Multimedia Conference*. Seattle, WA 9–13 November 1997, 183–92. New York: ACM Press.

Faraday, P., and A. G. Sutcliffe. 1998. Making contact points between text and images. In *Proceedings ACM Multimedia '98 of 6th ACM International Multimedia Conference*, 29–37. New York: ACM Press.

Fischer, G., E. Giaccardi, Y. Ye, A. G. Sutcliffe, and N. Mehandjiev. 2004. A framework for end-user development: Socio-technical perspectives and meta-design. *Commun ACM* 47(9):33–9.

Fogg, B. J. 1998. Persuasive computer: Perspectives and research directions. In *Proceedings: Human Factors in Computing Systems: CHI '98*, Los Angeles CA 18–23 April 1998, 225–32. New York: ACM Press.

Gardiner, M., and B. Christie. 1987. *Applying Cognitive Psychology to User Interface Design*. Chichester, UK: Wiley.

Hartmann, J., A. G. Sutcliffe, and A. De Angeli. 2008. Towards a theory of user judgment of aesthetics and user interface quality. *ACM Trans Comput Hum Interact* 15(4):15–30.

Hassenzahl, M. 2004. The interplay of beauty, goodness and usability in interactive products. *Hum Comput Interact* 19(4):319–49.

Hegarty, M., and M. A. Just. 1993. Constructing mental models of text and diagrams. *J Mem Lang* 32:717–42.

Hochberg, J. 1986. Presentation of motion and space in video and cinematic displays. In *Handbook of Perception and Human Performance, 1: Sensory Processes and Perception*, ed. K. R. Boff, L. Kaufman, and J. P. Thomas. New York: Wiley.

Hollan, J. D., E. L. Hutchins, and L. Weitzman. 1984. Steamer: An interactive inspectable simulation-based training system. *AI Mag* 5(2):15–27.

Hornof, A., and T. Halverson. 2003. Cognitive strategies and eye movements for searching hierarchical computer displays. In *CHI 2003 Conference Proceedings of Conference on Human Factors in Computing Systems*. New York: ACM Press.

ISO. 1997. *ISO 9241: Ergonomic Requirements for Office Systems with Visual Display Terminals (VDTs)*. Geneva, Switzerland: International Standards Organisation.

ISO. 1998. *ISO 14915 Multimedia User Interface Design Software Ergonomic Requirements, Part 1: Introduction and Framework; Part 3: Media Combination and Selection*. International Standards Organisation. Geneva, Switzerland.

Kristof, R., and A. Satran. 1995. *Interactivity by Design: Creating and Communicating with New Media*. Mountain View, CA: Adobe Press.

Lavie, T., and N. Tractinsky. 2004. Assessing dimensions of perceived visual aesthetics of web sites. *Int J Hum Comput Stud* 60(3):269–98.

Levie, W. H., and R. Lentz. 1982. Effects of text illustrations: A review of research. *Educ Comput Technol J* 30(4):159–232.

Lidwell, W., K. Holden, and J. Butler. 2003. *Universal Principles of Design*. Gloucester, MA: Rockport.

Lowe, D., and W. Hall. 1998. *Hypermedia and the Web*. Chichester, West Sussex: John Wiley.

Mann, W. C., and S. A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organisation. *Text* 8(3):243–81.

May, J., and P. Barnard. 1995. Cinematography and interface design. In *Proceedings: Fifth IFIP TC 13 International Conference on Human-Computer Interaction, Lillehammer, Norway 27–29 June 1995*, ed. K. Nordbyn, P. H. Helmersen, D. J. Gilmore, and S. A. Arnesen, 26–31. London: Chapman & Hall.

Mullet, K., and D. Sano. 1995. *Designing Visual Interfaces: Communication Oriented Techniques*. Englewood Cliffs, NJ: SunSoft Press.

Narayanan, N. H., and M. Hegarty. 1998. On designing comprehensible interactive hypermedia manuals. *Int J Hum Comput Stud* 48:267–301.

Nielsen, J. 1995. *Multimedia and Hypertext: The Internet and Beyond*. Boston, MA: AP Professional.

Norman, D. A. 2004. *Emotional Design: Why We Love (or Hate) Everyday Things*. New York: Basic Books.

Norman, D. A., and T. Shallice. 1986. Attention to action: Willed and automatic control of behaviour. In *Consciousness and Self-Regulation*, ed. G. E. Davidson and G. E. Schwartz, Vol. 4, 1–18. New York: Plenum.

Ortony, A., G. L. Clore, and A. Collins. 1988. *The Cognitive Structure of Emotions*. Cambridge: Cambridge University Press.

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Reeves, B., and C. Nass. 1996. *The Media Equation: How People Treat Computers, Television and New Media Like Real People and Places*. Stanford CA/Cambridge: CLSI/Cambridge University Press.

Rogers, Y., and M. Scaife. 1998. How can interactive multimedia facilitate learning? In *Intelligence and Multimodality in Multimedia Interfaces: Research and Applications*, ed. J. Lee. Menlo Park, CA: AAAI Press.

Scaife, M., Y. Rogers, F. Aldrich, and M. Davies. 1997. Designing for or designing with? Informant design for interactive learning environments. In *Proceedings: Human Factors in Computing Systems CHI '97*, Atlanta GA 22–27 May 1997, ed. S. Pemberton, 343–50. New York: ACM Press.

Spool, J. M., T. Scanlon, C. Snyder, W. Schroeder, and T. de Angelo. 1999. *Web Site Usability: A Designer's Guide*. San Francisco, CA: Morgan Kaufmann.

Sutcliffe, A. G. 1997. Task-related information analysis. *Int J Hum Comput Stud* 47(2):223–57.

Sutcliffe, A. G. 1999. User-centered design for multimedia applications. In *Proceedings Vol. 1: IEEE Conference on Multimedia Computing and Systems, Florence*, 116–23. Los Alamitos, CA: IEEE Computer Society Press.

Sutcliffe, A. G. 2002. Assessing the reliability of heuristic evaluation for website attractiveness and usability. In *Proceedings HICSS-35: Hawaii International Conference on System Sciences*, Hawaii 7–10 January 2002, 1838–47. Los Alamitos, CA: IEEE Computer Society Press.

Sutcliffe, A. G. 2003. *Multimedia and Virtual Reality: Designing Multisensory User Interfaces*. Mahwah, NJ: Lawrence Erlbaum Associates.

Sutcliffe, A. G. 2009. Designing for user engagement: Aesthetic and attractive user interfaces. In *Synthesis Lectures on Human Centered Informatics*, ed. J. M. Carroll. San Rafael, CA: Morgan Claypool.

Sutcliffe, A. G., and A. De Angeli. 2005. Assessing interaction styles in web user interfaces. In *Proceedings of Human Computer Interaction—Interact 2005*, 405–17. Berlin, Germany: Springer.

Sutcliffe, A. G., and P. Faraday. 1994. Designing presentation in multimedia interfaces. In *CHI '94 Conference Proceedings: Human Factors in Computing Systems 'Celebrating Interdependence,'* Boston, MA, April 24–28, 1994, ed. B. Adelson, S. Dumais, and J. Olson, 92–8. New York: ACM Press.

Tan, D. S., G. R. Robertson, and M. Czerwinski. 2001. Exploring 3D navigation: Combining speed coupled flying with orbiting. In *CHI 2001 Conference Proceedings: Conference on Human Factors in Computing Systems*, Seattle, March 31–April 5, 2001, ed. J. A. Jacko, A. Sears, M. Beaudouin-Lafon, and R. J. K. Jacob, 418–25. New York: ACM Press.

Tractinsky, N. 1997. Aesthetics and apparent usability: Empirically assessing cultural and methodological issues. In *Human Factors in Computing Systems: CHI '97 Conference Proceedings*, Atlanta GA May 22–27, 1997, ed. S. Pemberton, 115–22. New York: ACM Press.

Tractinsky, N., A. Shoval-Katz, and D. Ikar. 2000. What is beautiful is usable. *Interact Comput* 13(2):127–45.

Travis, D. 1991. *Effective Colour Displays: Theory and Practice*. Boston, MA: Academic Press.

Treisman, A. 1988. Features and objects: Fourteenth Bartlett memorial lecture. *Q J Exp Psychol* 40A(2):201–37.

Tufte, E. R. 1997. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CN: Graphics Press.

Wickens, C. D., D. Sandry, and M. Vidulich. 1983. Compatibility and resource competition between modalities of input, output and central processing. *Hum Factors* 25:227–48.

Zhou, M. X., and S. K. Feiner. 1998. Visual task characterization for automated visual discourse synthesis. In *Human Factors in Computing Systems of CHI '98 Conference Proceedings*. New York: ACM Press.

18 Multimodal Interfaces

Sharon Oviatt

CONTENTS

18.1 WHAT ARE MULTIMODAL SYSTEMS, AND WHY ARE WE BUILDING THEM?

Multimodal systems process two or more combined user input modes—such as speech, pen, touch, manual gestures, gaze, and head and body movements—in a coordinated manner with multimedia system output. This class of systems represents a new direction for computing, and a paradigm shift away from conventional Windows, Icons, Menus, and Pointer interfaces. Since the appearance of Bolt's (1980) "Put That There" demonstration system, which processed speech in parallel with touch pad pointing, a variety of new multimodal systems has emerged. This new class of interfaces aims to recognize naturally occurring forms of human language and behavior, which incorporate at least one recognition-based technology (e.g., speech, pen, vision). The development of novel multimodal systems has been enabled by the myriad input and output technologies currently becoming available, including new devices and improvements in recognition-based technologies. This chapter will review the main types of multimodal interfaces, their advantages and cognitive science underpinnings, primary features and architectural characteristics, and general research in the field of multimodal interaction and interface design.

The growing interest in multimodal interface design is inspired largely by the goal of supporting more transparent, flexible, efficient, and powerfully expressive means of human–computer interaction (HCI). Multimodal interfaces are expected to be easier to learn and use, and are preferred by users for many applications. They have the potential to expand computing to more challenging applications, to be used by a broader spectrum of everyday people, and to accommodate more adverse usage conditions than in the past. Such systems also have the potential to function in a more robust and stable manner than unimodal recognition systems involving a single recognition-based technology, such as speech, pen, or vision.

The advent of multimodal interfaces based on recognition of human speech, gaze, gesture, and other natural behavior represents only the beginning of a progression toward computational interfaces capable of relatively human-like sensory perception. Such interfaces eventually will interpret continuous input from a large number of different visual, auditory, and tactile input modes, which will be recognized as users engage in everyday activities. The same system will track and incorporate information from multiple sensors on the user's interface and surrounding physical environment in order to support intelligent adaptation to the user, task and usage environment. Future adaptive multimodal-multisensor interfaces have the potential to support new functionality, to achieve unparalleled robustness, and to perform flexibly as a multifunctional and personalized mobile system.

18.2 WHAT TYPES OF MULTIMODAL INTERFACES EXIST, AND WHAT IS THEIR HISTORY AND CURRENT STATUS?

Multimodal systems have developed rapidly during the past decade, with steady progress toward building more general and robust systems, as well as more transparent human interfaces than ever before (Benoit et al. 2000; Oviatt et al. 2000). Major developments have occurred in the hardware and software needed to support key component technologies incorporated within multimodal systems, as well as in techniques for integrating parallel input streams. Multimodal systems also have diversified to include new modality combinations, including speech and pen input, speech and lip movements, speech and manual gesturing, and gaze tracking and manual input (Benoit and Le Goff 1998; Cohen et al. 1997; Stork and Hennecke 1995; Turk and Robertson 2000; Zhai, Morimoto, and Ihde 1999). In addition, the array of multimodal applications has expanded extremely rapidly in recent years. Among other areas, it presently includes multimodal map-based systems for mobile and in-vehicle use, multimodal browsers, multimodal interfaces to virtual reality systems for simulation and training, multimodal person identification/verification systems for security purposes, multimodal medical, educational, robotics, military, and web-based transaction systems, and multimodal access and management of personal information on handhelds and cell phones (Cohen and McGee 2004; Iyengar, Nock, and Neti 2003; McGee 2003; Neti et al. 2000; Oviatt 2003; Oviatt, Flickner, and Darrell 2004; Oviatt and Lunsford 2005; Oviatt et al. 2000; Pankanti, Bolle, and Jain 2000; Reithinger et al. 2003).

In one of the earliest multimodal concept demonstrations, Bolt had users sit in front of a projection of "Dataland" in "the Media Room" (Negroponte 1978). Using the "Put That There" interface (Bolt 1980), they could use speech and pointing on an armrest-mounted touch pad to create and move objects on a two-dimensional (2D) large-screen display. For example, the user could issue a command to "Create a blue square there," with the intended location of "there" indicated by a 2D cursor mark on the screen. Semantic processing was based on the user's spoken input, and the meaning of the deictic "there" was resolved by processing the *x*,*y* coordinate indicated by the cursor at the time "there" was uttered. Since Bolt's early prototype, considerable strides have been made in developing a wide variety of different types of multimodal systems.

Among the earliest and most rudimentary multimodal systems were ones that supported speech input along with a standard keyboard-and-mouse interface. Conceptually, these multimodal interfaces represented the least departure from traditional graphical user interfaces (GUIs). Their initial focus was on providing richer natural language processing to support greater expressive power for the user when manipulating complex visuals and engaging in information extraction. As speech recognition technology matured during the late 1980s and 1990s, these systems added spoken input as an alternative to text entry with the keyboard. As such, they represent early involvement of the natural language and speech communities in developing the technologies needed to support new multimodal interfaces. Among the many examples of this type of multimodal interface are CUBRICON, Georal, Galaxy, XTRA, Shoptalk, and Miltalk (Cohen et al. 1989; Kobsa et al. 1986; Neal and Shapiro 1991; Seneff et al. 1996; Siroux et al. 1995; Wahlster 1991).

Several of these early systems were multimodal-multimedia map systems to which a user could speak or type and point with a mouse to extract tourist information or engage in military situation assessment (Cohen et al. 1989; Neal and Shapiro 1991; Seneff et al. 1996; Siroux et al. 1995). For example, using the CUBRICON system, a user could point to an object on a map and ask: *"Is this <point> an air base?"* CUBRICON was an expert system with extensive domain knowledge, as well as natural language-processing capabilities that included referent identification and dialog tracking (Neal and Shapiro 1991). With the Georal system, a user could query a tourist information system to plan travel routes using spoken input and pointing at a touch-sensitive screen (Siroux et al. 1995). In contrast, the Shoptalk system permitted users to interact with complex graphics representing factory production flow for chip manufacturing (Cohen et al. 1989). Using Shoptalk, a user could point to a specific machine in the production layout and issue the command: "Show me all the times when this machine was down." After the system delivered its answer as a list of time ranges, the user could click on one to ask the follow-up question, "What chips were waiting in its queue then, and were any of them hot lots?" Multimedia system feedback was available in the form of a text answer, or the user could click on the machine in question to view an exploded diagram of the machine queue's contents during that time interval.

More recent multimodal systems have moved away from processing simple mouse or touch pad pointing, and have begun designing systems based on two parallel input streams that each are capable of conveying rich semantic information. These multimodal systems recognize two natural forms of human language and behavior, for which two recognition-based technologies are incorporated within a more powerful bimodal user interface. To date, systems that combine either speech and pen input (Oviatt and Cohen 2000) or speech and lip movements (Benoit et al. 2000; Stork and Hennecke 1995; Rubin et al. 1998; Potamianos et al. 2003) constitute the two most mature areas within the field of multimodal research. In these cases, the keyboard and mouse have been abandoned. For speech and pen systems, spoken language sometimes is processed along with complex pen-based gestural input involving hundreds of different symbolic interpretations beyond pointing* (Oviatt et al. 2000). For speech and lip movement systems, spoken language is processed along with corresponding human lip movements during the natural audiovisual experience of spoken interaction. In both of these

* However, other recent pen/voice multimodal systems that emphasize mobile processing, such as multimodal interactive notepad and the Field Medic Information System (Holzman 1999; Huang et al. 2000), typically still limit pen input to pointing.

subliteratures, considerable work has been directed toward quantitative modeling of the integration and synchronization characteristics of the two rich input modes being processed, and innovative time-sensitive architectures have been developed to process these patterns in a robust manner.

Multimodal systems that recognize speech and pen-based gestures first were designed and studied in the early 1990s (Oviatt et al. 1992), with the original QuickSet system prototype built in 1994. The QuickSet system is an agent-based collaborative multimodal system that runs on a handheld PC (Cohen et al. 1997). With QuickSet, for example, a user can issue a multimodal command such as "Airstrips … facing this way <draws arrow>, and facing this way <draws arrow>," using combined speech and pen input to place the correct number, length, and orientation (e.g., SW, NE) of aircraft landing strips on a map. Other research-level systems of this type were built in the late 1990s. Examples include the Human-Centric Word Processor, Portable Voice Assistant, QuickDoc and MVIEWS (Bers, Miller, and Makhoul 1998; Cheyer 1998; Oviatt et al. 2000; Waibel et al. 1997). These systems represent a variety of different system features, applications, and information fusion and linguistic processing techniques. For illustration purposes, a comparison of five different speech and gesture systems is summarized in Figure 18.1. In most cases, these multimodal systems jointly interpreted speech and pen input based on a frame-based method of information fusion and a late semantic fusion approach, although QuickSet used a statistically-ranked unification process and a hybrid symbolic/statistical architecture (Wu, Oviatt, and Cohen 1999). Other recent systems also have adopted unification-based multimodal fusion and a hybrid architectural approach for processing multimodal input (Bangalore and Johnston 2000, 2009; Denecke and Yang 2000; Pfleger 2004; Wahlster 2001) and even multimodal output (Kopp, Tepper, and Cassell 2004).

Multimodal systems that process speech and continuous 3D manual gesturing are emerging, although these systems remain less mature than ones that process 2D pen input

Multimodal system characteristics	**QuickSet**	**Human-centric word processor**	**VR aircraft maintenance training**	**MATCH**	**Portable voice assistant**
Recognition of simultaneous or alternative individual modes	Simultaneous and individual modes	Simultaneous and individual modes	Simultaneous and individual modes	Simultaneous and individual modes	Simultaneous and individual modes
Type and size of gesture vocabulary	Pen input, multiple gestures, large vocabulary	Pen input, deictic selection	3D manual input, multiple gestures, small vocabulary	Pen input, multiple gestures, handwriting recognition, small vocabulary	Pen input, deictic selection, handwriting recognition
Size of speech vocabulary[1] and type of linguistic processing	Moderate vocabulary, grammar-based	Large vocabulary, statistical language processing	Small vocabulary, grammar-based	Large vocabulary, grammar-based	Small vocabulary, grammar-based
Type of signal fusion	Late semantic fusion, unification, hybrid symbolic/statistical, MTC framework	Late semantic fusion, frame-based	Late semantic fusion, frame-based	Late semantic fusion, unification, finite state transducers (lattice output)	Late semantic fusion, frame-based
Type of platform and applications	Wireless handheld, varied map and VR applications, digital paper	Desktop computer, word processing	Virtual reality system, aircraft maintenance training	Wireless handheld,[2] map-based city information (restaurants, movies, route planning)	Wireless handheld, catalog ordering
Evaluation status	Proactive user-centered design and iterative system evaluations	Proactive user-centered design	Planned for future	Proactive user-centered design derived from QuickSet and system language evaluation	Planned for future

[1] A small speech vocabulary is up to 200 words, moderate 300–1000 words, and large in excess of 1000 words. For pen-based gestures, deictic selection is an individual gesture, a small vocabulary is 2–20 gestures, moderate 20–100, and large in excess of 100 gestures.

[2] Kiosk version with a talking head is also available.

FIGURE 18.1 Examples of functionality, architectural features, and general classification of different speech and gesture multimodal applications.

(Encarnacao and Hettinger 2003; Flanagan and Huang 2003; Sharma, Pavlovic, and Huang 1998; Pavlovic, Sharma, and Huang 1997). This primarily is because of the significant challenges associated with segmenting and interpreting continuous manual movements, compared with a stream of *x,y* ink coordinates. As a result of this difference, multimodal speech and pen systems have advanced more rapidly in their architectures, and have progressed further toward commercialization of applications. However, a significant cognitive science literature is available for guiding the design of emerging speech and 3D gesture prototypes (Condon 1988; Kendon 1980; McNeill 1992), which will be discussed further in Section 18.5. Among the earlier systems to begin processing, manual pointing or 3D gestures combined with speech were those developed by Koons, Sparrell, and Thorisson (1993), Sharma et al. (1996), Poddar et al. (1998), and Duncan et al. (1999).

Historically, multimodal speech and lip movement research has been driven by cognitive science interest in intersensory audiovisual perception and the coordination of speech output with lip and facial movements (Benoit and Le Goff 1998; Bernstein and Benoit 1996; Cohen and Massaro 1993; Massaro and Stork 1998; McGrath and Summerfield 1985; McGurk and MacDonald 1976; McLeod and Summerfield 1987; Robert-Ribes et al. 1998; Sumby and Pollack 1954; Summerfield 1992; Vatikiotis-Bateson et al. 1996). Among the many contributions of this literature has been a detailed classification of human lip movements (visemes) and the viseme-phoneme mappings that occur during articulated speech. Existing systems that have processed combined speech and lip movements include the classic work by Petajan (1984), Brooke and Petajan (1986), and others (Adjoudani and Benoit 1995; Bregler and Konig 1994; Silsbee and Su 1996; Tomlinson, Russell, and Brooke 1996). Additional examples of speech and lip movement systems, applications, and relevant cognitive science research have been detailed elsewhere (Benoit et al. 2000). Researchers in this area have been actively exploring adaptive techniques for improving system robustness, especially in noisy environmental contexts (Dupont and Luettin 2000; Meier, Hürst, and Duchnowski 1996; Potamianos et al. 2003; Rogozan and Deglise 1998), which is an important future research direction. Although this literature has not emphasized the development of applications, nonetheless its quantitative modeling of synchronized phoneme/viseme patterns has been used to build animated characters that generate text-to-speech output and coordinated lip movements. These new animated characters are being used as an interface design vehicle for facilitating users' multimodal interaction with next-generation conversational interfaces (Cassell et al. 2000; Cohen and Massaro 1993).

While the main multimodal literatures to date have focused on either speech and pen input or speech and lip movements, recognition of other modes also is maturing and beginning to be integrated into new kinds of multimodal systems. In particular, there is growing interest in designing multimodal interfaces that incorporate vision-based technologies, such as interpretation of gaze, facial expressions, head nodding, gesturing, and large body movements (Flanagan and Huang 2003; Morency et al. 2005; Morimoto et al. 1999; Pavlovic, Berry, and Huang 1997; Turk and Robertson 2000; Zhai, Morimoto, and Ihde 1999). These technologies unobtrusively or *passively* monitor user behavior and need not require explicit user commands to a "computer." This contrasts with *active input modes*, such as speech or pen, which the user deploys intentionally as a command issued to the system (see Figure 18.2). Although passive modes may be "attentive" and less obtrusive, active modes generally are more reliable indicators of user intent.

As vision-based technologies mature, one important future direction will be the development of *blended* multimodal interfaces that combine both passive and active modes. These interfaces typically will be *temporally cascaded*, so one goal in designing new prototypes will be to determine optimal processing strategies for using advance information from the first mode (e.g., gaze) to constrain accurate interpretation of the following modes (e.g., gesture, speech). This kind of blended multimodal interface potentially can provide users with greater transparency and control, while also supporting improved robustness and broader application functionality (Oviatt and Cohen 2000; Zhai, Morimoto, and Ihde 1999). As this collection of technologies matures, there also is strong interest in designing new types of pervasive and mobile interfaces, including ones capable of adaptive processing to the user and environmental context.

As multimodal interfaces gradually evolve toward supporting more advanced recognition of users' natural activities in context, they will expand beyond rudimentary bimodal systems to ones that incorporate three or more input modes, qualitatively different modes, and more sophisticated models of multimodal interaction. This trend already has been initiated within biometrics research, which has combined recognition of multiple behavioral input modes (e.g., voice, handwriting) with physiological ones (e.g., retinal scans, fingerprints) to achieve reliable person identification and verification in challenging field conditions (Choudhury et al. 1999; Jain et al. 1999; Jain and Ross 2002; Pankanti, Bolle, and Jain 2000).

Related to passive multimodal interfaces and biometrics, in recent years considerable international multimodal research has focused on the collection and automatic analysis of audiovisual activity patterns during collaborative group meetings. This work included hardware, software, and interface-level research toward improving the robustness of information capture and interpretation using multimodal techniques and machine learning. These projects have included CALO in North America (http://caloproject.sri .com/), the IM2 and AMIDA projects in Europe (http://www.im2.ch/), and others. Although much of this research was aimed at engineering-level advances in detection and analysis of human activity patterns related to surveillance and similar applications, nonetheless other aspects were focused on designing computational meeting assistants for copresent and remote interactions (Lunsford, Oviatt & Arthur, 2006; Lunsford, Oviatt & Coulston, 2005; Oviatt, Swindells, and Arthur 2008), multimodal meeting browsers (Lalanne et al. 2005; Lisowska 2007; Popescu-Belis and Georgescul 2006; Popescu-Belis et al. 2008; Whittaker, *Multimodal interfaces* process two or more combined user input modes—such as speech, pen, touch, manual gestures, gaze, and head and body movements—in a coordinated manner with multimedia system output. They are a new class of interfaces that aim to recognize naturally occurring forms of human language and behavior, and which incorporate one or more recognition-based technologies (e.g., speech, pen, vision).

Active input modes are ones that are deployed by the user intentionally as an explicit command to a computer system (e.g., speech).

Passive input modes refer to naturally occurring user behavior or actions that are recognized by a computer (e.g., facial expressions, manual gestures). They involve user input that is unobtrusively and passively monitored, without requiring any explicit command to a computer.

Blended multimodal interfaces are ones that incorporate system recognition of at least one passive and one active input mode (e.g., speech and lip movement systems).

Temporally cascaded multimodal interfaces are ones that process two or more user modalities that tend to be sequenced in a particular temporal order (e.g., gaze, gesture, speech), such that partial information supplied by recognition of an earlier mode (e.g., gaze) is available to constrain interpretation of a later mode (e.g., speech). Such interfaces may combine only active input modes, only passive ones, or they may be blended.

Mutual disambiguation involves disambiguation of signal or semantic-level information in one error-prone input mode from partial information supplied by another. Mutual disambiguation can occur in a multimodal architecture with two or more semantically rich recognition-based input modes. It leads to recovery from unimodal recognition errors within a multimodal architecture, with the net effect of suppressing errors experienced by the user.

Simultaneous integrator refers to a user who habitually presents two input signals (e.g., speech, pen) in a temporally overlapped manner when communicating multimodal commands to a system.

Sequential integrator refers to a user who habitually separates his or her multimodal signals, presenting one before the other with a brief pause intervening.

Multimodal hypertiming refers to the fact that both sequential and simultaneous integrators will further accentuate their basic multimodal integration pattern when under duress (e.g., as task difficulty or system recognition errors increase).

Visemes refers to the detailed classification of visible lip movements that correspond with consonants and vowels during articulated speech. A *viseme-phoneme mapping* refers to the correspondence between visible lip movements and audible phonemes during continuous speech.

Feature-level fusion is a method for fusing low-level feature information from parallel input signals within a multimodal architecture, which has been applied to processing closely synchronized input such as speech and lip movements.

Semantic-level fusion is a method for integrating semantic information derived from parallel input modes in a multimodal architecture, which has been used for processing speech and gesture input.

Frame-based integration is a pattern-matching technique for merging attribute-value data structures to fuse semantic information derived from two input modes into a common meaning representation during multimodal language processing.

Unification-based integration is a logic-based method for integrating partial meaning fragments derived from two input modes into a common meaning representation during multimodal language processing. Compared with frame-based integration, unification derives from logic programming, and has been more precisely analyzed and widely adopted within computational linguistics.

FIGURE 18.2 Multimodal interface terminology.

Laban, and Tucker 2005), and adaptive human-centered interfaces for collaborative work, problem solving, and educational exchanges (Barthelmess and Oviatt 2007; Lunsford, Oviatt & Arthur, 2006; Lunsford, Oviatt & Coulston, 2005; Oviatt, Swindells, and Arthur 2008). Archivus is one example of a multimodal meeting browser and retrieval system, which users could query with speech, keyboard and mouse, or pen-based natural language queries to obtain information about meeting events (Lisowska 2007). Other research on group multimodal activity patterns established new approaches to detecting floor control, agreement, and decisions during conversation, dominance of group members, and other interaction dynamics (Chen and Harper 2009; Gatica-Perez et al. 2005; Germesin 2009; Hsueh and Moore 2008).

Apart from these developments within research-level systems, multimodal interfaces also are being commercialized as products, especially in areas like personal information access and management on handhelds and cell phones. Microsoft's handheld multimodal interactive notepad (MiPad) for personal information management, (Huang et al. 2000), Kirusa's cell phone interface for directory assistance and messaging, and AT&T's iMOD for searching movies on the iPhone (Johnston 2009) are just three examples of the many mobile commercial products that are being developed with multimodal interfaces. They include spoken language processing and a stylus or touch input for selecting fields to constrain and guide the natural language processing. In some cases, keyboard input is supported as a third option, as well as multimedia output in the form of visualizations and text-to-speech. Another visible growth area for multimodal interfaces involves in-vehicle control of navigation, communication, and entertainment systems, which have emerged in both domestic and import cars. Mobile map-based systems and systems for safety-critical medical and military applications also are being commercialized by companies like Adapx (Adapx, 2011). Which places an emphasis on developing tangible multimodal interfaces that preserve users' existing work practice, minimize cognitive load, and provide backups in case of system failure (Cohen and McGee 2004; McGee 2003). Section 18.9 provides further discussion of trends in the commercialization of multimodal interfaces.

18.3 WHAT ARE THE GOALS AND ADVANTAGES OF MULTIMODAL INTERFACE DESIGN?

Over the past decade, numerous advantages of multimodal interface design have been documented. Unlike a traditional keyboard-and-mouse interface or a unimodal recognition-based interface, multimodal interfaces permit flexible use of input modes. This includes the choice of which modality to use for conveying different types of information, to use combined input modes, or to alternate between modes at any time. Because individual input modalities are well suited in some situations, and less ideal or even inappropriate in others, modality choice is an important design issue in a multimodal system. As systems become more complex and multifunctional, a single modality simply does not permit all users to interact effectively across all tasks and environments.

Because there are large individual differences in ability and preference to use different modes of communication, a multimodal interface permits diverse user groups to exercise selection and control over how they interact with the computer (Fell et al. 1994; Karshmer and Blattner 1998). In this respect, multimodal interfaces have the potential to accommodate a broader range of users than traditional interfaces—including users of different ages, skill levels, native language status, cognitive styles, sensory impairments, and other temporary illnesses or permanent handicaps. For example, a visually impaired user or one with repetitive stress injury may prefer speech input and text-to-speech output. In contrast, a user with a hearing impairment or accented speech may prefer touch, gesture, or pen input. The natural alternation between modes that is permitted by a multimodal interface also can be effective in preventing overuse and physical damage to any single modality, especially during extended periods of computer use (Markinson,* personal communication, 1993).

Multimodal interfaces also provide the adaptability that is needed to accommodate the continuously changing conditions of mobile use. In particular, systems involving speech, pen, or touch input are suitable for mobile tasks and, when combined, users can shift among these modalities from moment to moment as environmental conditions change (Holzman 1999; Oviatt 2000b,c). There is a sense in which mobility can induce a state of temporary disability, such that a person is unable to use a particular input mode for some period of time. For example, the user of an in-vehicle application may frequently be unable to use manual or gaze input, although speech is relatively more available. In this respect, a multimodal interface permits the modality choice and switching that is needed during the changing environmental circumstances of actual field and mobile use.

A large body of data documents that multimodal interfaces satisfy higher levels of user preference when interacting with simulated or real computer systems. Users have a strong preference to interact multimodally, rather than unimodally, across a wide variety of different application domains, although this preference is most pronounced in spatial domains (Hauptmann 1989; Oviatt 1997). For example, 95%–100% of users preferred to interact multimodally, when they were free to use either speech or pen input in a map-based spatial domain (Oviatt 1997). During pen/voice multimodal interaction, users preferred speech input for describing objects and events, sets and subsets of objects, out-of-view objects, conjoined information, past and future temporal states, and for issuing commands for actions or iterative actions (Cohen and Oviatt 1995; Oviatt and Cohen 1991). However, their preference for pen input increased when conveying digits, symbols, graphic content, and especially when conveying the location and form of spatially-oriented information on a dense graphic display such as a map (Oviatt and Olsen 1994; Oviatt 1997; Suhm 1998). Likewise, 71% of users combined speech and manual gestures multimodally, rather than using one input mode, when manipulating graphic objects on a cathode ray tube screen (Hauptmann 1989).

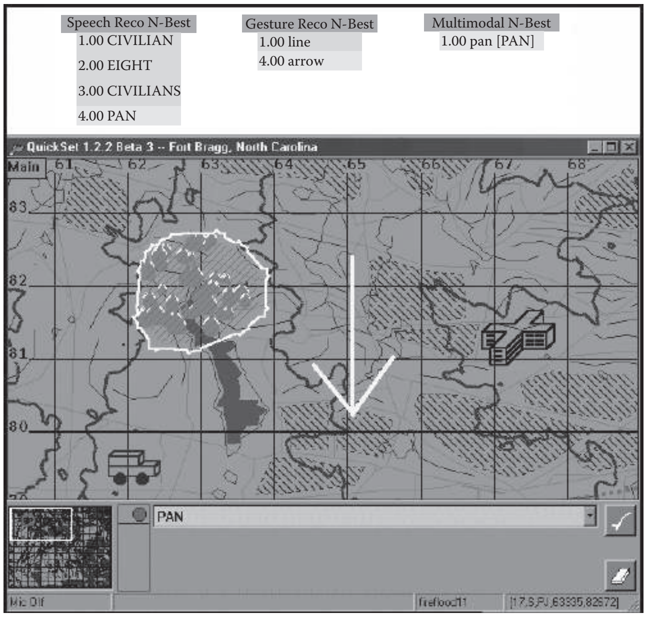
During the early design of multimodal systems, it was assumed that efficiency gains would be the main advantage of designing an interface multimodally, and that this advantage would derive from the ability to process input modes in parallel. It is true that multimodal interfaces sometimes support improved efficiency, especially when manipulating graphical information. In simulation research comparing speech-only with multimodal pen/voice interaction, empirical work demonstrated that multimodal interaction yielded 10% faster task completion time during visual-spatial tasks, but no significant efficiency advantage in verbal or quantitative task domains (Oviatt 1997; Oviatt, Cohen, and Wang 1994). Likewise, users' efficiency improved when they combined speech and gestures multimodally to manipulate 3D objects, compared with unimodal input (Hauptmann 1989). In another early study, multimodal speech and mouse input improved efficiency in a line-art drawing task (Leatherby and Pausch 1992). Finally, in a study that compared task completion times for a graphical interface versus a multimodal pen/voice interface, military domain experts averaged four times faster at setting up complex simulation scenarios on a map when they were able to interact multimodally (Cohen, McGee, and Clow 2000). This latter study was based on testing of a fully functional multimodal system, and it included time required to correct recognition errors.

One particularly advantageous feature of multimodal interface design is its superior error handling, both in terms of error avoidance and graceful recovery from errors (Oviatt and van Gent 1996; Oviatt, Bernard, and Levow 1999; Oviatt 1999a; Rudnicky and Hauptmann 1992; Suhm 1998; Tomlinson, Russell, and Brooke 1996). There are user-centered and system-centered reasons why multimodal systems facilitate error recovery, when compared with unimodal recognition-based interfaces. For example, in a multimodal speech and pen-based gesture interface, users will select the input mode that they judge to be less error prone for particular lexical content, which tends to lead to error avoidance (Oviatt and van Gent 1996). They may prefer speedy speech input, but will switch to pen input to communicate a foreign surname. Second, users' language often is simplified when interacting multimodally, which can substantially reduce the complexity of natural language processing and thereby reduce recognition errors (Oviatt and Kuhn 1998; see Section 18.5 for discussion). In one study, users' multimodal utterances were documented to be briefer, to contain

* R. Markinson, University of California at San Francisco Medical School, 1993.

fewer complex locative descriptions, and 50% fewer spoken disfluencies, when compared with a speech-only interface. Third, users have a strong tendency to switch modes after system recognition errors, which facilitates error recovery. This error resolution occurs because the confusion matrices differ for any given lexical content for the different recognition technologies involved in processing (Oviatt, Bernard, and Levow 1999; Oviatt 2002).

In addition to these user-centered reasons for better error avoidance and resolution, there also are system-centered reasons for superior error handling. A well-designed multimodal architecture with two semantically rich input modes can support *mutual disambiguation* of input signals. For example, if a user says "ditches" but the speech recognizer confirms the singular "ditch" as its best guess, then parallel recognition of several graphic marks can result in recovery of the correct plural interpretation. This recovery can occur in a multimodal architecture even though the speech recognizer initially ranks the plural interpretation "ditches" as a less-preferred choice on its n-best list. Mutual disambiguation involves recovery from unimodal recognition errors within a multimodal architecture because semantic information from each input mode supplies partial disambiguation of the other mode, thereby leading to more stable and robust overall system performance (Oviatt 1999a, 2000a, 2002). Another example of mutual disambiguation is shown in Figure 18.3. To achieve optimal error handling, a multimodal interface ideally should be designed to include complementary input modes, and also the alternative input modes provide duplicate functionality such that users can accomplish their goals using either mode.



Speech Reco N-Best	Gesture Reco N-Best	Multimodal N-Best
1.00 CIVILIAN	1.00 line	1.00 pan [PAN]
2.00 EIGHT	4.00 arrow	
3.00 CIVILIANS		
4.00 PAN		

PAN

FIGURE 18.3 **(See color insert.)** Multimodal command to "pan" the map, which illustrates mutual disambiguation occurring between incoming speech and gesture information, such that lexical hypotheses were pulled up on both n-best lists to produce a correct final multimodal interpretation.

In two recent studies involving more than 4600 multimodal commands, a multimodal architecture was found to support mutual disambiguation and error suppression ranging between 19% and 41% (Oviatt 1999a, 2000a, 2002). Improved robustness also was greater for "challenging" user groups (accented vs. native speakers) and usage contexts (mobile vs. stationary use). These results indicate that a well-designed multimodal system not only can perform more robustly than a unimodal system, but also in a more stable way across varied real-world users and usage contexts. Finally, during audiovisual perception of speech and lip movements, improved speech recognition also has been demonstrated for both human listeners (McLeod and Summerfield 1987) and multimodal systems (Adjoudani and Benoit 1995; Tomlinson, Russell, and Brooke 1996).

Another recent focus has been on the advantages of multimodal interface design for minimizing users' cognitive load. As task complexity increases, there is evidence that users self manage their working memory limits by distributing information across multiple modalities, which in turn enhances their task performance during both perception and production (Calvert, Spence, and Stein 2004; Mousavi, Low, and Sweller 1995; Oviatt 1997; Oviatt, Flickner, and Darrell 2004; Tang et al. 2005). These predictions and findings are based on Wickens and colleagues' cognitive resource theory and Baddeley's theory of working memory (Baddeley 1992; Wickens, Sandry, and Vidulich 1983). The latter maintains that short-term or working memory consists of multiple independent processors associated with different modes. This includes a visual-spatial "sketch pad" that maintains visual materials such as pictures and diagrams in one area of working memory, and a separate phonological loop that stores auditory-verbal information. Although these two processors are believed to be coordinated by a central executive, in terms of lower-level modality processing, they are viewed as functioning largely independently, which is what enables the effective size of working memory to expand when people use multiple modalities during tasks (Baddeley 1992). So with respect to management of cognitive load, the inherent flexibility of multimodal interfaces is well suited to accommodating the high and changing load conditions typical of realistic mobile use.

One natural human communication mode, pen input, recently has been demonstrated to support better problem solving in science, technology, engineering, mathematics domains than existing keyboard-based graphical interfaces. In two separate studies, when the same high school students solved the same mathematics and science problems, the best performance was supported by a digital pen and paper interface, followed by a pen tablet interface, and lastly a keyboard-based graphical interface (Oviatt, Arthur, and Cohen 2006; Oviatt and Cohen 2010a). Performance advantages due to the pen interface included improved speed, ability to focus attention, solve problems correctly, engage in high-level synthetic thinking, communicate fluently, and remember information just solved (Oviatt 2006; Oviatt, Arthur, and Cohen 2006; Oviatt et al. 2007). In addition, low-performing students experienced elevated cognitive load, with tablet interfaces disrupting their performance more than high-performers, which in turn expanded the achievement gap between groups (Oviatt, Arthur, and Cohen 2006).

Other recent findings have revealed that pen input, a rich communications modality that is capable of expressing different representations (linguistic, symbolic, diagrammatic, numeric), can facilitate people's communicative fluency, ideational fluency, and problem solving to a greater extent than either a keyboard-based graphical interface or nondigital paper and pencil (Oviatt and Cohen, 2010b; Oviatt in press). When biology students, working on hypothesis-generation tasks were compared using different interfaces, they demonstrated greater *nonlinguistic communicative fluency* (numeric, symbolic, diagrammatic) when using pen interfaces than a graphical interface or paper and pencil. In parallel, pen interfaces stimulated 36% more ideational fluency (i.e., appropriate biology hypothesis generation), and regression analyses revealed that interface support for nonlinguistic fluency predicted ideational fluency, as shown in Figure 18.4a (Oviatt in press). In contrast, a keyboard-based graphical interface elicited more *linguistic fluency*, which suppressed ideation, as shown in Figure 18.4b (Oviatt in press). These results challenge us to question the adequacy of existing keyboard-centric graphical interfaces for future educational and professional practice. Pen input capabilities, which often are included in multimodal interfaces, effectively provided a single focused input tool for *fluently expressing both nonlinguistic and linguistic representations*, and for *shifting rapidly and flexibly among them* without impeding thought (Oviatt and Cohen, 2010b; Oviatt in press).

18.4 WHAT METHODS AND INFORMATION HAVE BEEN USED TO DESIGN NOVEL MULTIMODAL INTERFACES?

The design of new multimodal systems has been inspired and organized largely by two things. First, the cognitive science literature on intersensory perception and intermodal coordination during production is beginning to provide a foundation of information for user modeling, as well as information on what systems must recognize and how multimodal architectures should be organized. For example, the cognitive science literature has provided knowledge of the natural integration patterns that typify people's lip and facial movements with speech output (Benoit et al. 1996; Ekman 1992; Ekman and Friesen 1978; Fridlund 1994; Hadar et al. 1983; Massaro and Cohen 1990; Stork and Hennecke 1995; Vatikiotis-Bateson et al. 1996), and their coordinated use of manual or pen-based gestures with speech (Kendon 1980; McNeill 1992; Oviatt, DeAngeli, and Kuhn 1997). Given the complex nature of users' multimodal interaction, cognitive science has and will continue to play an essential role in guiding the design of robust multimodal systems. In this respect, a multidisciplinary perspective will be more central to successful multimodal system design than it has been for traditional GUI design. The cognitive science underpinnings of multimodal system design are described in Section 18.5.

Second, high-fidelity automatic simulations also have played a critical role in prototyping new types of multimodal systems (Dahlbäck, Jëonsson, and Ahrenberg 1992; Oviatt et al. 1992). When a new multimodal system is in the planning stages, design sketches, and low-fidelity mock-ups may initially be used to visualize the new system and plan the sequential flow of HCI. These tentative design plans then are rapidly transitioned into a

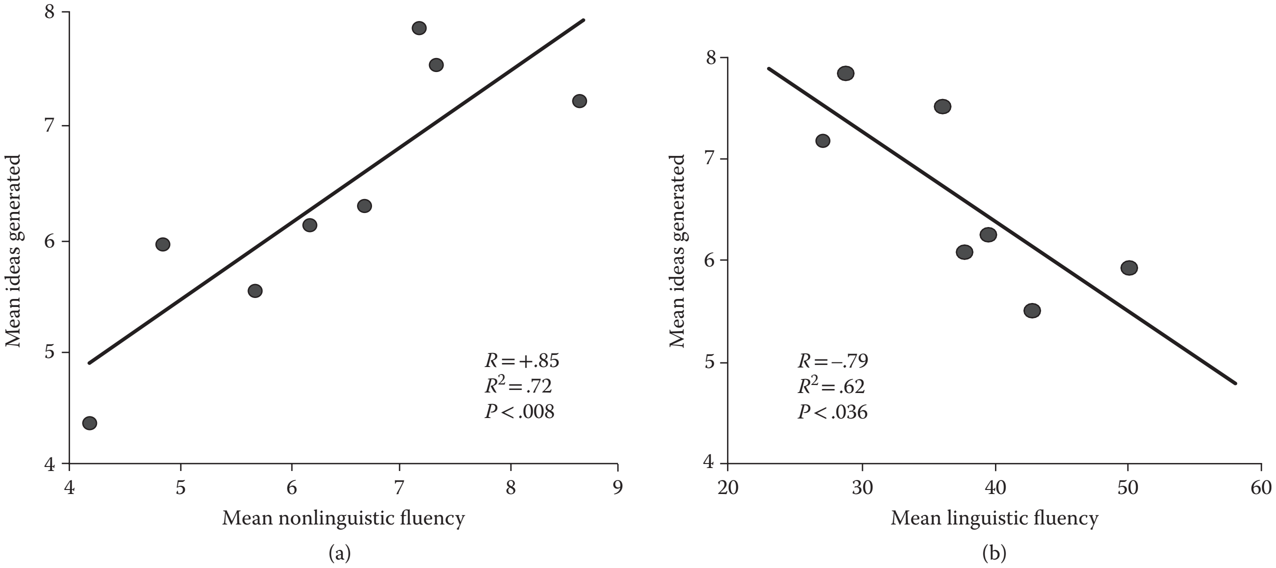


FIGURE 18.4 (a) Regression analysis confirming *positive predictive relation* between interface support for nonlinguistic communicative fluency and ideational fluency as a result of using a richly expressive pen interface. (b) Regression confirming *negative predictive relation* between interface support for linguistic communicative fluency and ideational fluency when using a keyboard-based graphical interface.

higher-fidelity simulation of the multimodal system, which is available for proactive and situated data collection with the intended user population. High-fidelity simulations have been the preferred method for designing and evaluating new multimodal systems, and extensive data collection with such tools preferably is completed before a fully functional system is ever built.

During high-fidelity simulation testing, a user interacts with what she believes is a fully functional multimodal system, although the interface is actually a simulated front end designed to appear and respond as the fully functional system would. During the interaction, a programmer assistant at a remote location provides the simulated system responses. As the user interacts with the front end, the programmer tracks her multimodal input and provides system responses as quickly and accurately as possible. To support this role, the programmer makes use of automated simulation software that is designed to support interactive speed, realism with respect to the targeted system, and other important characteristics. For example, with these automated tools, the programmer may be able to make a single selection on a workstation field to rapidly send simulated system responses to the user during a data-collection session.

High-fidelity simulations have been the preferred method for prototyping multimodal systems for several reasons. Simulations are relatively easy and inexpensive to adapt, compared with building and iterating a complete system. They also permit researchers to alter a planned system's characteristics in major ways (e.g., input and output modes available), and to study the impact of different interface features in a systematic and scientific manner (e.g., type and base-rate of system errors). In comparison, a particular system with its fixed characteristics is a less flexible and suitable research tool, and the assessment of any single system basically amounts to an individual case study. Using simulation techniques, rapid adaptation and investigation of planned system features permits researchers to gain a broader and more principled perspective on the potential of newly emerging technologies. In a practical sense, simulation research can assist in the evaluation of critical performance tradeoffs and in making decisions about alternative system designs, which designers must do as they strive to create more usable multimodal systems.

The most recent high-fidelity simulation tools have been designed to collect data and prototype new multimodal systems that support collaborative group interactions (Arthur et al. 2006). They also are beginning to support real-time processing of the paralinguistic aspects of users' natural speech and pen input signals, such as changes in user amplitude that indicate intended addressee during multiperson exchanges, which is needed to develop new adaptive multimodal systems (Cohen et al. 2008; Oviatt, Swindells, and Arthur 2008). An example of a dual-wizard, high-fidelity simulation environment designed to prototype collaborative multimodal interfaces and also adapt to changes in users' speech and pen amplitude is shown in Figure 18.5. This particular simulation collected speech, visual, and digital pen and paper data from students during three-person collaborative meetings while they used a computational assistant to solve mathematics problems. Two wizards were required in this simulation to process key user data in real time involving the (1) linguistic content of users' requests, and (2) amplitude of their speech or pen communication. Specialized simulation software and wizard training both were needed to support adequately fast and error-free teamwork between the two wizards (Cohen et al. 2008). To support the further development and commercialization of multimodal systems, additional infrastructure that will be needed in the future includes the following: (1) simulation tools for rapidly building and reconfiguring multimodal interfaces, (2) automated tools for collecting and analyzing multimodal corpora, and (3) automated tools for iterating new multimodal systems to improve their performance (see Oviatt et al. 2000, for further discussion).

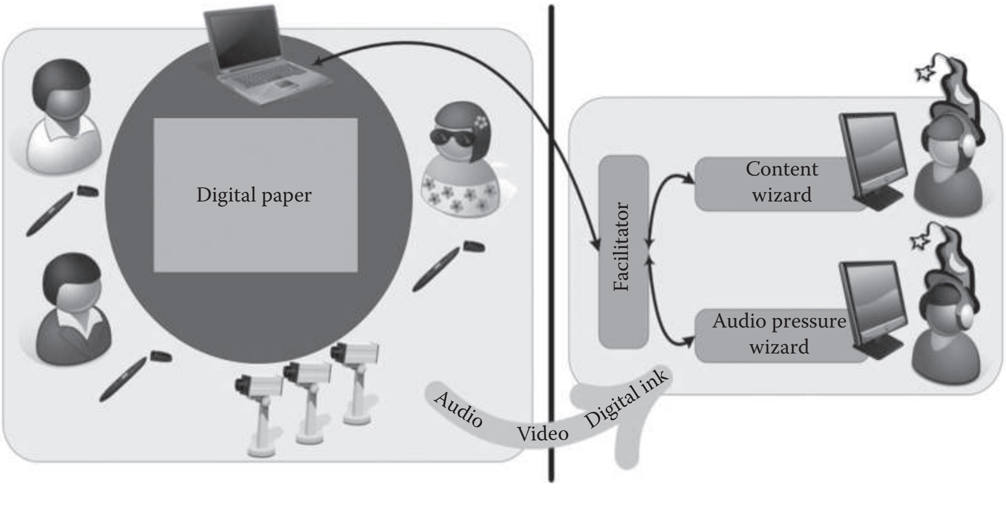


FIGURE 18.5 Dual-wizard simulation environment for prototyping collaborative multimodal interfaces capable of real-time adaptive processing.

18.5 WHAT ARE THE COGNITIVE SCIENCE UNDERPINNINGS OF MULTIMODAL INTERFACE DESIGN?

This section discusses the growing cognitive science literature that provides the empirical underpinnings needed to design next-generation multimodal interfaces. The ability to develop multimodal systems depends on knowledge of the natural integration patterns that typify people's combined use of different input modes. In particular, the design of new multimodal systems depends on intimate knowledge of the properties of different modes and the information content they carry, the unique characteristics of multimodal language and its processability, and the integration and synchronization characteristics of users' multimodal interaction. It also relies on accurate prediction of when users are likely to interact multimodally, and how alike different users are in their specific integration patterns. The relevant cognitive science literature on these topics is very extensive, especially when consideration is given to all of the underlying sensory perception and production capabilities involved in different input modes currently being incorporated in new multimodal interfaces. As a result, this section will be limited to introducing the main cognitive science themes and findings that are relevant to the more common types of multimodal systems.

This cognitive science foundation also has played a key role in identifying computational "myths" about multimodal interaction, and replacing these misconceptions with contrary empirical evidence. Figure 18.6 summarizes 10 common myths about multimodal interaction, which are addressed and discussed in more detail elsewhere (Oviatt 1999b). As such, the literature summarized in this section aims to provide a more accurate foundation for guiding the design of next-generation multimodal systems.

18.5.1 When Do Users Interact Multimodally?

During natural interpersonal communication, people are always interacting multimodally. Of course, in this case, the number of information sources or modalities that an interlocutor has available to monitor is essentially unlimited. However, all multimodal systems are constrained in the number and type of input modes they can recognize. Also, a user can compose active input during human–computer interaction that either is delivered multimodally or that is delivered entirely using just one mode. That is, although users in general may have a strong preference to interact multimodally rather than unimodally, this is no guarantee that they will issue every command to a system multimodally, given the particular type of multimodal interface available. Therefore, the first nontrivial question that arises during system processing is whether a user is communicating unimodally or multimodally.

In the case of speech and pen-based multimodal systems, users typically intermix unimodal and multimodal expressions. In one study involving a visual-spatial domain, users' commands were expressed multimodally 20% of the time, with others just spoken or written (Oviatt et al. 1997). In contrast, in spatial domains, the ratio of users' multimodal interaction often is 65%–70% (Oviatt, Bernard, and Levow 1999). Predicting whether a user will express a command multimodally also depends on the type of action she is performing. In particular, users almost always express commands multimodally when describing spatial information about the location, number, size, orientation, or shape of an object. In one study, users issued multimodal commands 86% of the time when they had to add, move, modify, or calculate the distance between objects on a map in a way that required specifying spatial locations (Oviatt et al. 1997). They also were moderately likely to interact multimodally when selecting an object from a larger array, for example, when deleting a particular object from the map. However, when performing general actions without any spatial component, such as printing a map, users expressed themselves multimodally less than 1% of the time. These data emphasize that future multimodal systems will need to distinguish between instances when users are and are not communicating multimodally, so that accurate decisions can be made about when parallel input streams should be interpreted jointly versus individually. They also suggest that knowledge of the type of actions to be included in an application, such as whether the application entails manipulating spatial information, should influence the basic decision of whether to build a multimodal interface at all.

Myth #1: *If you build a multimodal system, users will interact multimodally.*
Myth #2: *Speech and pointing is the dominant multimodal integration pattern.*
Myth #3: *Multimodal input involves simultaneous signals.*
Myth #4: *Speech is the primary input mode in any multimodal system that includes it.*
Myth #5: *Multimodal language does not differ linguistically from unimodal language.*
Myth #6: *Multimodal integration involves redundancy of content between modes.*
Myth #7: *Individual error-prone recognition technologies combine multimodally to produce even greater unreliability.*
Myth #8: *All users' multimodal commands are integrated in a uniform way.*
Myth #9: *Different input modes are capable of transmitting comparable content.*
Myth #10: *Enhanced efficiency is the main advantage of multimodal systems.*

FIGURE 18.6 Ten myths of multimodal interaction: separating myth from empirical reality.

Findings from a more recent study reveal that multimodal interface users spontaneously respond to dynamic changes in their own cognitive load by shifting to multimodal communication as load increases with task difficulty and communicative complexity (Oviatt, Coulston, and Lunsford 2004). Given a flexible multimodal interface, users' ratio of multimodal (vs. unimodal) interaction increased substantially from 18.6% when referring to established dialog context to 77.1% when required to establish a new context, a +315% relative increase. Likewise, the ratio of users' multimodal interaction increased significantly as the tasks became more difficult, from 59.2% during low-difficulty tasks to 65.5% at moderate difficulty, 68.2% at high difficulty, and 75.0% at very high difficulty, an overall relative increase of +27%. These adaptations in multimodal interaction levels reflect users' effort to self manage limitations in their working memory as discourse-level demands and task complexity increased. As discussed earlier, they accomplished this by distributing communicative information across multiple modalities in a manner compatible with a cognitive load theory of multimodal interaction. This interpretation is consistent with Baddeley's theory of working memory (Baddeley 1992), as well as the growing literatures within education (Mousavi, Low, and Sweller 1995; Sweller 1988), linguistics (Almor 1999), and multisensory perception (Calvert, Spence, and Stein 2004; Ernst and Bulthoff 2004). Recent work on visual and haptic processing under workload also indicates that presentation of haptic feedback during a complex task can augment users' ability to handle visual information overload (Tang et al. 2005).

In a multimodal interface that processes passive or blended input modes, there always is at least one passively-tracked input source providing continuous information (e.g., gaze tracking, head position). In these cases, all user input would by definition be classified as multimodal, and the primary problem would become segmentation and interpretation of each continuous input stream into meaningful actions of significance to the application. In the case of blended multimodal interfaces (e.g., gaze tracking and mouse input), it still may be opportune to distinguish active forms of user input that might be more accurately or expeditiously handled as unimodal events.

18.5.2 What Are the Integration and Synchronization Characteristics of Users' Multimodal Input?

The past literature on multimodal systems has focused largely on simple selection of objects or locations in a display, rather than considering the broader range of multimodal integration patterns. Because the development of Bolt's (1980) "Put That There" system, speak-and-point has been viewed as the prototypical form of multimodal integration. In Bolt's system, semantic processing was based on spoken input, but the meaning of a deictic term such as "that" was resolved by processing the *x*, *y* coordinate indicated by pointing at an object. Since that time, other multimodal systems also have attempted to resolve deictic expressions using a similar approach, for example, using gaze location instead of manual pointing (Koons et al. 1993).

Unfortunately, this concept of multimodal interaction as point-and-speak makes only limited use of new input modes for *selection* of objects—just as the mouse does. In this respect, it represents the persistence of an old mouse-oriented metaphor. In contrast, modes that transmit written input, manual gesturing, and facial expressions are capable of generating symbolic information that is much more richly expressive than simple pointing or selection. In fact, studies of users' integrated pen/voice input indicate that a speak-and-point pattern only comprises 14% of all spontaneous multimodal utterances (Oviatt et al. 1997). Instead, pen input more often is used to create graphics, symbols and signs, gestural marks, digits, and lexical content. During interpersonal multimodal communication, linguistic analysis of spontaneous manual gesturing also indicates that simple pointing accounts for less than 20% of all gestures (McNeill 1992). Together, these cognitive science and user-modeling data highlight the fact that any multimodal system designed exclusively to process speak-and-point will fail to provide users with much useful functionality. For this reason, specialized algorithms for processing deictic-point relations will have only limited practical use in the design of future multimodal systems. It is clear that a broader set of multimodal integration issues needs to be addressed in future work. Future research also should explore typical integration patterns between other promising modality combinations, such as speech and gaze.

It also is commonly assumed that any signals involved in a multimodal construction will co-occur temporally. The presumption is that this temporal overlap then determines which signals to combine during system processing. In the case of speech and manual gestures, successful processing of the deictic term "that square" in Bolt's original system relied on interpretation of pointing when the word "that" was spoken in order to extract the intended referent. However, one empirical study indicated that users often do not speak deictic terms at all, and when they do the deictic frequently is not overlapped in time with their pointing. In fact, it has been estimated that as few as 25% of users' commands actually contain a spoken deictic that overlaps with the pointing needed to disambiguate its meaning (Oviatt et al. 1997).

Beyond the issue of deixis, a series of studies has shown that users' input frequently does not overlap at all during multimodal commands to a computer (Oviatt 1999b; Oviatt et al. 2003; Xiao, Girand, and Oviatt 2002; Xiao et al. 2003). In fact, there are two distinct types of user with respect to integration patterns—*simultaneous* integrators and *sequential* ones. A user who habitually integrates his or her speech and pen input in a *simultaneous* manner overlaps them temporally, whereas a sequential integrator finishes one mode before beginning the second. These two types of user integration pattern occur across the lifespan from children through the elderly (Oviatt, Lunsford et al. 2005; Xiao, Girand, and Oviatt 2002; Xiao et al. 2003). They also can be detected almost immediately during multimodal interaction, usually on the very first input. Users' habitual integration pattern remains strikingly highly consistent during a session, as well as resistant to change following explicit instructions or attempts at training (Oviatt et al. 2003; Oviatt, Lunsford et al. 2005). This bimodal distribution of user integration patterns has been observed in different task domains (e.g., map-based real estate selection, crisis management, educational applications with animated characters), and also when using different types of interface (e.g., conversational, command style) (Darves and Oviatt 2004; Oviatt 1999b; Xiao, Girand, and Oviatt 2002; Xiao et al. 2003). In short, empirical studies have demonstrated that this bimodal distinction between users in their fundamental integration pattern generalizes widely across different age groups, task domains, and types of interface.

One interesting discovery in recent work is the phenomenon of *multimodal hypertiming*, which refers to the fact that both sequential and simultaneous integrators will entrench further or accentuate their habitual multimodal integration pattern (i.e., increasing their intermodal *lag* during sequential integrations, or *overlap* during simultaneous integrations) during system error handling or when completing increasingly difficult tasks. In fact, users will progressively increase their degree of entrenchment by 18% as system errors increase and by 59% as task difficulty increases (Oviatt et al. 2003). As such, changes in the degree of users' multimodal hypertiming provides a potentially sensitive means of evaluating their cognitive load during real-time interactive exchanges. In the context of system error handling, the phenomenon of multimodal hypertiming basically replaces the hyperarticulation that is typically observed in users during error-prone speech-only interactions.

Given the bimodal distribution of user integration patterns, *adaptive temporal thresholds* potentially could support more tailored and flexible approaches to fusion. Ideally, an adaptive multimodal system would detect, automatically learn, and adapt to a user's dominant multimodal integration pattern, which could result in substantial improvements in system processing speed, the accuracy of interpretation, and also synchronous interchange with the user. For example, it has been estimated that system delays could be reduced to approximately 40%–50% of what they currently are by adopting user-defined thresholds (Oviatt, Lunsford et al. 2005; Gupta 2004). Recent research has begun comparing different learning-based models for adapting a multimodal system's temporal thresholds to an individual user in real time (Huang and Oviatt 2006).

Unfortunately, users' multimodal integration patterns have not been studied as extensively or systematically for other input modes, such as speech and manual gesturing. Linguistics research on interpersonal communication patterns has revealed that both spontaneous gesturing and signed language often precede their spoken lexical analogs during human communication (Kendon 1980; Naughton 1996), when considering word-level integration pattern. In fact, the degree to which gesturing precedes speech is greater in topic-prominent languages such as Chinese than it is in subject-prominent ones like Spanish or English (McNeill 1992). Even in the speech and lip movement literature, close but not perfect temporal synchrony is typical, with lip movements occurring a fraction of a second before the corresponding auditory signal (Abry, Lallouache, and Cathiard 1996; Benoit 2000). However, when considering the whole user utterance as the unit of analysis, some other studies of speech and manual gesturing have found a higher rate of simultaneity for these modes (Epps, Oviatt, and Chen 2004). Learning-based approaches that are capable of accurately identifying and adapting to different multimodal integration patterns, whether due to differences among users, modality combinations, or applications and usage contexts, will be required in order to generalize and speed up multimodal system development in the future.

In short, although two input modes may be highly interdependent and synchronized during multimodal interaction, synchrony does not imply simultaneity. The empirical evidence reveals that multimodal signals often do not co-occur temporally at all during human–computer or natural human communication. Therefore, multimodal system designers cannot necessarily count on conveniently overlapped signals in order to achieve successful processing in the multimodal architectures they build. Future research needs to explore the integration patterns and temporal cascading that can occur among three or more input modes, such as gaze, gesture, and speech, so that more advanced multimodal systems can be designed and prototyped.

In the design of new multimodal architectures, it is important to note that data on the order of input modes and average time lags between input modes has been used to determine the likelihood that an utterance is multimodal versus unimodal, and to establish temporal thresholds for fusion of input. In the future, weighted likelihoods associated with different utterance segmentations, for example, that an input stream containing speech, writing, speech should be segmented into [S / W S] rather than [S W / S], and with intermodal time lag distributions, will be used to optimize correct recognition of multimodal user input (Oviatt 1999b). In the design of future time-critical multimodal architectures, data on users' integration and synchronization patterns will need to be collected for other mode combinations during realistic interactive tasks, so that temporal thresholds can be established for performing multimodal fusion.

18.5.3 What Individual Differences Exist in Multimodal Interaction, and What Are the Implications for Designing Systems for Universal Access?

There are large individual differences in users' multimodal interaction patterns, beginning with their overall preference to interact unimodally versus multimodally, and also which mode they generally prefer (e.g., speaking vs. writing) (Oviatt, Coulston, and Lunsford 2004). As outlined in Section 18.5.2, there likewise are striking differences among users' in adopting either a sequential or simultaneous multimodal integration pattern. Recent research has revealed that these two patterns are associated with behavioral and linguistic differences between the groups (Oviatt, Lunsford et al. 2005). Although in an interactive task context their performance speed was comparable, sequential integrators were far less error prone and excelled during new or complex tasks. Although their speech rate was no slower, sequential integrators also had more precise articulation (e.g., fewer disfluencies). Finally, sequential integrators were more likely to adopt terse and direct command-style language, with a smaller and less varied vocabulary, which appeared focused on achieving error-free communication. These user differences in interaction pattern have been interpreted as deriving from fundamental differences among users in their reflective-impulsive cognitive style (Oviatt, Lunsford et al. 2005). Based on this work, one goal of future multimodal interface design will be to support the poorer attention span and higher error rate of impulsive users—especially for mobile in-vehicle, military, and similar application contexts in which the cost of committing errors is unacceptably high.

Apart from these individual differences, cultural differences also have been documented between users in modality integration patterns. For example, substantial individual differences have been reported in the temporal synchrony between speech and lip movements (Kricos 1996) and, in addition, lip movements during speech production are known to be less exaggerated among Japanese speakers than Americans (Sekiyama and Tohkura 1991). In fact, extensive interlanguage differences have been observed in the information available from lip movements during audiovisual speech (Fuster-Duran 1996). These findings have implications for the degree to which disambiguation of speech can be achieved through lip movement information in noisy environments or for different user populations. Finally, non-native speakers, the hearing impaired, and elderly listeners all are more influenced by visual lip movement than auditory cues when processing speech (Fuster-Duran 1996; Massaro 1996). These results have implications for the design and expected value of audiovisual multimedia output for different user groups in animated character interfaces. With respect to support for universal access, recent work also has shown the advantage of combined audiovisual processing for recognition of impaired speech (Potamianos and Neti 2001).

Finally, gender, age, and other individual differences are common in gaze patterns, as well as speech and gaze integration (Argyle 1972). As multimodal interfaces incorporating gaze become more mature, further research will need to explore these gender and age-specific patterns, and to build appropriately adapted processing strategies. In summary, considerably more research is needed on multimodal integration and synchronization patterns for new mode combinations, as well as for diverse and disabled users for whom multimodal interfaces may be especially suitable for ensuring universal access.

18.5.4 Is Complementarity or Redundancy the Main Organizational Theme That Guides Multimodal Integration?

It frequently is claimed that the propositional content conveyed by different modes during multimodal communication contains a high degree of redundancy. However, the dominant theme in users' natural organization of multimodal input actually is complementarity of content, not redundancy. For example, speech and pen input consistently contribute different and complementary semantic information—with the subject, verb, and object of a sentence typically spoken, and locative information written (Oviatt et al. 1997). In fact, a major complementarity between speech and manually oriented pen input involves visual-spatial semantic content, which is one reason these modes are an opportune combination for visual-spatial applications. Although spatial information is uniquely and clearly indicated through pen input, the strong descriptive capabilities of speech are better suited for specifying temporal and other nonspatial information. Even during multimodal correction of system errors, when users are highly motivated to clarify and reinforce their information delivery, speech and pen input express redundant information less than 1% of the time. Finally, during interpersonal communication, linguists also have documented that spontaneous speech and manual gesturing involve complementary rather than duplicate information between modes (McNeill 1992).

Other examples of primary multimodal complementarities during interpersonal and human–computer communication have been described in past research (McGurk and MacDonald 1976; Oviatt and Olsen 1994; Wickens, Sandry, and Vidulich 1983). For example, in the literature on multimodal speech and lip movements, natural feature-level complementarities have been identified between visemes and phonemes for vowel articulation, with vowel rounding better conveyed visually, and vowel height and backness better revealed auditorally (Massaro and Stork 1998; Robert-Ribes et al. 1998).

In short, actual data highlight the importance of complementarity as a major organizational theme during multimodal communication. Furthermore, recent research has documented an increase in the ratio of complementary to redundant multimodal constructions as users' cognitive load increases (Ruiz, Oviatt, and Chen, 2010). The designers of next-generation multimodal systems therefore should not expect to rely on duplicated information when processing multimodal language, although in certain contexts such as teaching a higher percentage of redundant content may exist because of the tutorial context.

In multimodal systems involving both speech and pen-based gestures and speech and lip movements, one explicit goal has been to integrate complementary modalities in a manner that yields a synergistic blend, such that each mode can be capitalized upon and used to overcome weaknesses in the other mode (Cohen et al. 1989). This approach to system design has promoted the philosophy of using modes and component technologies to their natural advantage, and of combining them in a manner that permits mutual disambiguation. One advantage of achieving such a blend is that the resulting multimodal architecture can function more robustly than an individual recognition-based technology or a multimodal system based on input modes lacking natural complementarities.

18.5.5 What Are the Primary Features of Multimodal Language?

Communication channels can be tremendously influential in shaping the language transmitted within them. From past research, there now is cumulative evidence that many linguistic features of multimodal language are qualitatively very different from that of spoken or formal textual language. In fact, it can differ in features as basic as brevity, semantic content, syntactic complexity, word order, disfluency rate, degree of ambiguity, referring expressions, specification of determiners, anaphora, deixis, and linguistic indirectness. In many respects, multimodal language is simpler linguistically than spoken language. In particular, comparisons have revealed that the same user completing the same map-based task communicates significantly fewer words, briefer sentences, and fewer complex spatial descriptions and also disfluencies when interacting multimodally, compared with using speech alone (Oviatt 1997). One implication of these findings is that multimodal interface design has the potential to support more robust future systems than a unimodal design approach. The following is an example of a typical user's spoken input while attempting to designate an open space using a map system: *"Add an open space on the north lake to b—include the north lake part of the road and north."* In contrast, the same user accomplished the same task multimodally by encircling a specific area and saying: *"Open space."*

In previous research, hard-to-process disfluent language has been observed to decrease by 50% during multimodal interaction with a map, compared with a more restricted speech-only interaction (Oviatt 1997). This drop occurs mainly because people have difficulty speaking spatial information, which precipitates disfluencies. In a flexible multimodal interface, they instead use pen input to convey spatial information, thereby avoiding the need to speak it. Further research is needed to establish whether other forms of flexible multimodal communication also generally ease users' cognitive load, which may be reflected in a reduced rate of disfluencies.

During multimodal pen/voice communication, the linguistic indirection that is typical of spoken language frequently is replaced with more direct commands (Oviatt and Kuhn 1998). In the following example, a study participant made a disfluent indirect request using speech input while requesting a map-based distance calculation: "What is the distance between the Victorian Museum and the, uh, the house on the east side of Woodpecker Lane?" When requesting distance information multimodally, the same user encircled the house and museum while speaking the following brief direct command: "Show distance between here and here." In this research, the briefer and more direct multimodal pen/voice language also contained substantially fewer referring expressions, with a selective reduction in coreferring expressions that instead were transformed into deictic expressions. This latter reduction in coreference would simplify natural language processing by easing the need for anaphoric tracking and resolution in a multimodal interface. Also consistent with fewer referring expressions, explicit specification of definite and indefinite reference is less common in multimodal language (Oviatt and Kuhn 1998). Current natural language-processing algorithms typically rely heavily on the specification of determiners in definite and indefinite references in order to represent and resolve noun phrase reference. One unfortunate by-product of the lack of such specifications is that current language-processing algorithms are unprepared for the frequent occurrence of elision and deixis in multimodal HCI.

In other respects, multimodal language clearly is different than spoken language, although not necessarily simpler. For example, users' multimodal pen/voice language departs from the canonical English word order of S-V-O-LOC (i.e., subject-verb-object-locative constituent), which is observed in spoken language and also formal textual language. Instead, users' multimodal constituents shift to an LOC-S-V-O word order. A recent study reported that 95% of locative constituents were in sentence-initial position during multimodal interaction. However, for the same users completing the same tasks while speaking, 96% of locatives were in sentence-final position (Oviatt et al. 1997). It is likely that broader analysis of multimodal communication patterns, which could involve gaze and manual gesturing to indicate location rather than pen-based pointing, would reveal a similar reversal in word order.

One implication of these many differences is that new multimodal corpora, statistical language models, and natural language-processing algorithms will need to be established before multimodal language can be processed optimally. Future research and corpus collection efforts also will be needed on different types of multimodal communication, and in other application domains, so that the generality of previously identified multimodal language differences can be explored.

18.6 WHAT ARE THE BASIC WAYS IN WHICH MULTIMODAL INTERFACES DIFFER FROM GRAPHICAL USER INTERFACES?

Multimodal research groups currently are rethinking and redesigning basic user interface architectures because a whole new range of architectural requirements has been posed. First, GUIs typically assume that there is a single event stream that controls the underlying event loop, with any processing sequential in nature. For example, most GUIs ignore typed input when a mouse button is depressed. In contrast, multimodal interfaces typically can process continuous and simultaneous input from parallel incoming streams. Second, GUIs assume that the basic interface actions, such as selection of an item, are atomic and unambiguous events. In contrast, multimodal systems process input modes using recognition-based technologies, which are designed to handle uncertainty and entail probabilistic methods of processing. Third, GUIs often are built to be separable from the application software that they control, although the interface components usually reside centrally on one machine. In contrast, recognition-based user interfaces typically have larger computational and memory requirements, which often makes it desirable to distribute the interface over a network so that separate machines can handle different recognizers or databases. For example, cell phones and networked PDAs may extract features from speech input, but transmit them to a recognizer that resides on a server. Finally, multimodal interfaces that process two or more recognition-based input streams require time-stamping of input, and the development of temporal constraints on mode fusion operations. In this regard, they involve uniquely time-sensitive architectures. As discussed in Section 18.5.2, recent research has been working toward the development of adaptive rather than fixed temporal thresholds, which can be tailored to specific modalities or a given user (Oviatt, Lunsford et al. 2005; Huang and Oviatt 2006). Adaptive temporal thresholds have not yet been implemented, but they could substantially improve multimodal processing speed, system usability, and portability of general processing modules across different types of multimodal systems.

18.7 WHAT BASIC ARCHITECTURES AND PROCESSING TECHNIQUES HAVE BEEN USED TO DEVELOP MULTIMODAL SYSTEMS?

Many early multimodal interfaces that handled combined speech and gesture, such as Bolt's "Put That There" system (Bolt 1980), have been based on a control structure in which multimodal integration occurs during the process of parsing spoken language. As discussed in Section 18.5.2, when the user speaks a deictic expression such as "here" or "this," the system searches for a synchronized gestural act that designates the spoken referent. Although such an approach is viable for processing a point-and-speak multimodal integration pattern, multimodal systems must be able to process richer input than just pointing, including gestures, symbols, graphic marks, lip movements, meaningful facial expressions, and so forth. To support more broadly functional multimodal systems, general processing architectures have been developed since Bolt's time. Some of these recent architectures handle a variety of multimodal integration patterns, as well as the interpretation of both unimodal and combined multimodal input. This kind of architecture can support the development of multimodal systems in which modalities are processed individually as input alternatives to one another, or those in which two or more modes are processed as combined multimodal input.

For multimodal systems designed to handle joint processing of input signals, there are two main subtypes of multimodal architecture. First, there are ones that integrate signals at the *feature level* (i.e., "early fusion") and others that integrate information at a *semantic level* (i.e., "late fusion"). Examples of systems based on an early feature-fusion processing approach include those developed by Bregler et al. (1993), Vo et al. (1995), and Pavlovic, Berry, and Huang (1997, 1998). In a feature-fusion architecture, the signal-level recognition process in one mode influences the course of recognition in the other. Feature fusion is considered more appropriate for closely temporally synchronized input modalities, such as speech and lip movements (Stork and Hennecke 1995; Rubin et al. 1998).

In contrast, multimodal systems using the late semantic fusion approach have been applied to processing multimodal speech and pen input or manual gesturing, for which the input modes are less closely coupled temporally. These input modes provide different but complementary information that typically is integrated at the utterance level. Late semantic integration systems use individual recognizers that can be trained using unimodal data, which are easier to collect and already are publicly available for speech and handwriting. In this respect, systems based on semantic fusion can be scaled up easier in number of input modes or vocabulary size. Examples of systems based on semantic fusion include Put That There (Bolt 1980), Shoptalk (Cohen et al. 1989), QuickSet (Cohen et al. 1997), CUBRICON (Neal and Shapiro 1991), Virtual World (Codella et al. 1992), Finger-Pointer (Fukumoto, Suenaga, and Mase 1994), VisualMan (Wang 1995), Human-Centric Word Processor, Portable Voice Assistant (Bers et al. 1998), the VR Aircraft Maintenance Training System (Duncan et al. 1999), Jeanie (Vo and Wood 1996), and MATCH (Bangalore and Johnston 2009).

As an example of multimodal information processing flow in a late-stage semantic architecture, Figure 18.7 illustrates two input modes (e.g., speech and manual or pen-based gestures) recognized in parallel and processed by an understanding component. The results involve partial meaning representations that are fused by the multimodal integration component, which also is influenced by the system's dialog management and interpretation of current context. During the integration process, alternative lexical candidates for the final multimodal interpretation are ranked according to their probability estimates on an n-best list. The best-ranked multimodal interpretation then is sent to the application invocation and control component, which transforms this information into a series of commands to one or more back-end application systems. System feedback typically includes multimedia output, which may incorporate text-to-speech and nonspeech audio, graphics and animation, and so forth. For examples of feature-based multimodal processing flow and architectures, especially as applied to multimodal speech and lip movement systems, see Benoit et al. (2000).

There are many ways to realize this information processing flow as an architecture. One common infrastructure that has been adopted by the multimodal research community involves *multiagent architectures*, such as the open agent

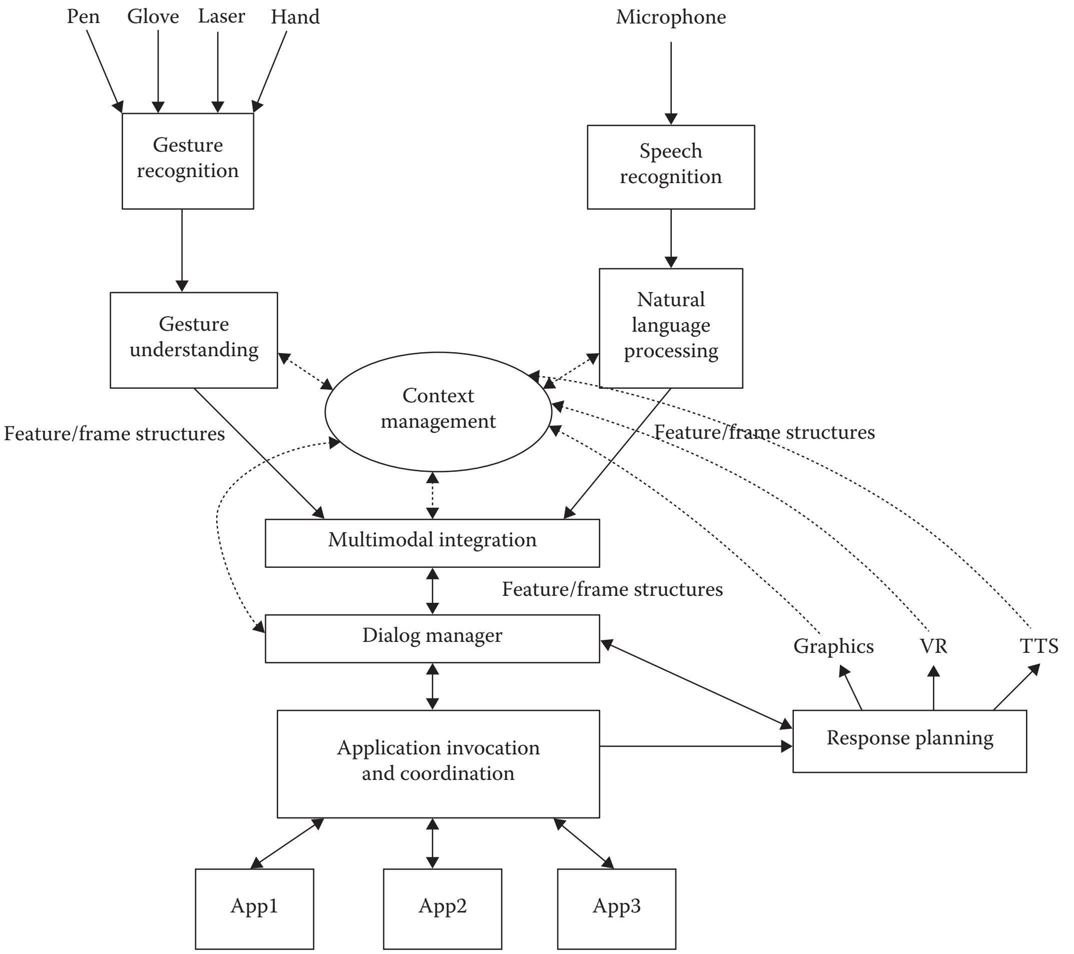


FIGURE 18.7 Typical information processing flow in a multimodal architecture designed for speech and gesture.

architecture (Cohen et al. 1994; Martin, Cheyer, and Moran 1999) and adaptive agent architecture (Kumar and Cohen 2000). In a multiagent architecture, the many components needed to support the multimodal system (e.g., speech recognition, gesture recognition, natural language processing, multimodal integration) may be written in different programming languages, on different machines, and with different operating systems. Agent communication languages are being developed that can handle asynchronous delivery, triggered responses, multicasting and other concepts from distributed systems, and that are fault-tolerant (Kumar and Cohen 2000). Using a multiagent architecture, for example, speech and gestures can arrive in parallel or asynchronously through individual modality agents, with the results recognized and passed to a facilitator. These results, typically an n-best list of conjectured lexical items and related time stamp information, then are routed to appropriate agents for further language processing. Next, sets of meaning fragments derived from the speech and pen signals arrive at the multimodal integrator. This agent decides whether and how long to wait for recognition results from other modalities, based on the system's temporal thresholds. It fuses the meaning fragments into a semantically- and temporally-compatible whole interpretation before passing the results back to the facilitator. At this point, the system's final multimodal interpretation is confirmed by the interface, delivered as multimedia feedback to the user, and executed by any relevant applications. In summary, multiagent architectures provide essential infrastructure for coordinating the many complex modules needed to implement multimodal system processing, and they permit doing so in a distributed manner that is compatible with the trend toward mobile computing.

When statistical processing techniques are combined with a symbolic unification-based approach that merges feature structures, then the multimodal architecture that results is a *hybrid symbolic/statistical* one. Hybrid architectures represent one major new direction for multimodal system development. Multimodal architectures also can be hybrids in the sense of combining Hidden Markov Models and Neural Networks. New hybrid architectures potentially are capable of achieving very robust functioning, compared with either an early- or late-fusion approach alone. For example, the Members-Teams-Committee hierarchical recognition technique, which is a hybrid symbolic/statistical multimodal integration framework trained over a labeled multimodal corpus, recently achieved 95.26% correct recognition performance, or within 1.4% of the theoretical system upper bound (Wu, Oviatt, and Cohen 1999). Other architectural approaches and contributions to processing multimodal information have been summarized elsewhere (Oliver and Horvitz 2005; Potamianos et al. 2003). Finally, in recent years, machine learning techniques have been applied to several aspects of multimodal systems, including individual modality recognition, early or late modality fusion, dialog management, and recognition and adaptation to users' multimodal communication integration patterns. This research has been spearheaded largely by European Union projects, and reported in a series of workshops on machine learning for multimodal interaction (Bengio 2004; Popescu-Belis, Renals, and Bourlard 2008).

18.8 WHAT BASIC LANGUAGE-PROCESSING TECHNIQUES AND STANDARDS HAVE BEEN USED TO DEVELOP MULTIMODAL SYSTEMS?

The core of multimodal systems based on semantic fusion involves algorithms that integrate common meaning representations derived from speech, gesture, and other modalities into a combined final interpretation. The semantic fusion operation requires a common meaning representation framework for all modalities, and a well-defined operation for combining partial meanings that arrive from different signals. To fuse information from different modalities, various research groups have independently converged on a strategy of recursively matching and merging attribute/value data structures, although using a variety of different algorithms (Vo and Wood 1996; Cheyer and Julia 1995; Pavlovic and Huang 1998; Shaikh et al. 1997). This approach is considered a *frame-based integration* technique. An alternative logic-based approach derived from computational linguistics (Carpenter 1990, 1992; Calder 1987) involves the use of *typed feature structures* and *unification-based integration*, which is a more general and well-understood approach. Unification-based integration techniques also have been applied to multimodal system design (Cohen et al. 1997; Johnston et al. 1997; Wu, Oviatt, and Cohen 1999). Feature-structure unification is considered well suited to multimodal integration because unification can combine complementary or redundant input from both modes, but it rules out contradictory input.

The unification-based approaches to multimodal language processing recently have been extended using finite-state transducers with lattice output (Bangalore and Johnston 2009). Multimodal language processing using these methods involve a one-stage interleaved grammar-processing approach, rather than separating speech parsing and multimodal integration. They also enable gesture aggregation, visual parsing, and more flexible and declarative encoding of temporal and spatial constraints (Bangalore and Johnston 2009). This approach, used in the MATCH system (see Figure 18.1), is efficient and enables tight coupling of more complex speech and gesture processing. Basically, the grammar is directly compiled into a cascade of finite-state transducers, which can compose with lattices from speech recognition and gesture recognition components (Bangalore and Johnston 2009). However, more research still is needed on the development of canonical meaning representations that are common among different input modes, which will need to be represented in new types of multimodal systems.

Because developing new multimodal interfaces remains a complex and specialized task, attention has turned to the establishment of W3C EMMA standards to provide a representation language for input to multimodal systems. This focus aims to facilitate plug-and-play components and rapid prototyping of new multimodal interfaces (Johnston 2009; Johnston et al. 2009). The new W3C EMMA standard addresses this problem by providing a standardized XML representation language for encapsulating and annotating input to multimodal systems. EMMA provides mechanisms for capturing and annotating the various stages of input processing. There are two key aspects to the language: a series of elements (e.g., emma:group, emma:one-of, emma:interpretation) that are used as containers for interpretations of the user input, and a series of annotation attributes and elements that are used to provide pieces of metadata associated with those inputs, such as time stamps (emma:start, emma:end) and confidence score values (emma:confidence). These standards currently are being used to develop new mobile applications, such as iMOD movies on demand for the iPhone and iMATCH for map-based searches of local restaurants (Johnston 2009). Both of these mobile systems involve processing of speech input with touch selection or pen gestures.

18.9 WHAT ARE THE TRENDS IN COMMERCIALIZING MULTIMODAL SYSTEMS?

Multimodal interfaces are a major and innovative departure from existing keyboard-and-mouse graphical interfaces. They represent a long-term interface redesign agenda and are challenging to build for many reasons outlined in this chapter. Section 18.3 provides a detailed summary of what is at stake in pushing forward to develop the next level of multimodal interfaces, and the advantages are overwhelmingly compelling. For this reason, multimodal interfaces have emerged during the last decade as a major worldwide trend for funding agencies, journal special issues, and also commercialization of new products.

The primary impetus for developing commercial multimodal interfaces to date has been accommodation of the practical aspects of usability while people are mobile and in field contexts, where a conventional keyboard interface is a poor or untenable option. To date, most of these commercialized multimodal interfaces have multimodal input-processing capabilities that are limited to (1) one natural and expressive mode (e.g., speech) plus simple pointing or selection in a second mode (e.g., touch or pen, as on Microsoft's MiPad), and (2) one keyboard-based input mode, and a second more natural and expressive mode (e.g., touch-based gestures, as on Apple's iPhone and iPad). Processing in these cases either takes place on alternative input modes individually, or the pointing or selection act simply constrains information processing involving the richer input mode, but requires no real fusion of the two information sources as discussed in Sections 18.6 and 18.7.

In other cases, current commercially available multimodal interfaces provide multimodal output, while input remains unimodal (e.g., animated characters with lifelike synchronized visual movements and text-to-speech output, now used in television commercials, kiosks, on the web, and elsewhere). In other cases, the multimodal input involves synchronized digital ink with recorded speech, but no coprocessing of the language content from these modes (e.g., Livescribe's digital pen). In Livescribe's case, processing is focused on recognition of written content in users' notes so it can be retrieved from past notes (Liverscribe, 2011). Temporally-associated verbatim audio recordings captured at the same time also can be played back.

Together, these illustrations highlight that commercially available multimodal interfaces primarily have been developed for mobile use, including cell phones, small PDA handhelds, and new digital pens. Second, they have avoided coprocessing and interpreting the linguistic meaning of two or more natural input streams. In this regard, they lag substantially behind far more powerful research-level prototypes, and have yet to reach their most valuable commercial potential. Third, in some cases, these systems simply have emphasized capture and reuse of synchronized human communication signals (e.g., verbatim speech, pen ink), rather than interpretation and processing of linguistic meaning at all. Finally, these system illustrations highlight the diverse meanings of what has been called "multimodal," which in recent years has become well recognized as a marketing strategy and advantage.

18.10 WHAT ARE THE MAIN FUTURE DIRECTIONS FOR MULTIMODAL INTERFACE DESIGN?

The computer science community is just beginning to understand how to design well-integrated and robust multimodal systems. To date, most multimodal systems remain bimodal, and recognition technologies related to several human senses (e.g., haptics, smell, taste) have yet to be well represented within multimodal interfaces. The successful design and development of new types of systems that include such modes will not be achievable through intuition. Rather, it will continue to require guidance from cognitive science on the coordinated human perception and production of natural modalities. In this respect, multimodal systems only can flourish through multidisciplinary cooperation, as well as teamwork among those representing expertise in the component technologies. In the future, further theoretical work also will be needed to account more coherently for diverse types of multimodal interaction patterns. New or refined theoretical frameworks could be invaluable for proactively guiding the design of new multimodal interfaces that are compatible with human capabilities and limitations. Current work in cognitive neuroscience and multisensory perception are beginning to provide an empirical and theoretical basis for this future interface design (Calvert, Spence, and Stein 2004; Ernst and Bulthoff 2004).

Most of the systems outlined in this chapter have been built during the past 15 years, and they are research-level systems. However, in some cases, they have developed well beyond the prototype stage, and are being integrated with other software at academic and federal sites, or appearing as newly shipped products. To achieve wider commercialization of multimodal interfaces, such systems will need to develop more powerful and general methods of natural language and dialog processing, and also temporal modeling and processing of incoming signals. In addition, multimodal data sets and tools are very much needed to build applications more rapidly in a wide range of domains, including for newly emerging collaborative multimodal applications such as meeting support and education (Barthelmess et al. 2005; Cohen and McGee 2004; Danninger et al. 2005; Gatica-Perez et al. 2005; McGee 2003; Pentland 2005). The many mobile multimodal interfaces currently being built also will require active adaptation to the user, task, ongoing dialog, and environmental context, which is another area of recent work (Gorniak and Roy 2005; Gupta 2004; Huang and Oviatt 2006; Jain and Ross 2002; Lunsford, Oviatt & Arthur, 2006; Lunsford, Oviatt & Coulston, 2005; Potamianos et al. 2003; Xiao et al. 2003). To facilitate the speed and generality of multimodal interface adaptation to these important variables, future work will need to integrate new machine learning techniques that are now being developed to handle asynchronous and heterogeneous data (Bengio 2004; McCowan et al. 2005).

New multimodal interfaces are just now beginning to appear on platforms such as collaborative table surfaces (Brandl et al. 2008; Leitner et al. 2009; Liwicki and El-Neklawy 2009). Tabletop interfaces previously have included multitouch input and virtual keyboards, but the latest versions now include more flexible and expressively powerful multimodal touch and write capabilities incorporating Anoto-based pen input (Anoto, 2009). Given these developments, high-resolution pen input now can be used on collaborative interfaces for marking, drawing, or writing directly on displayed documents, photographs, or simulations. This substantially expands possibilities for application functionality in education and other areas hoping to benefit from collaborative table interfaces. Another very recent change is the emergence of mobile digital book interfaces. They also are beginning to incorporate pen-based annotation capabilities for active learning and information reuse, along with graphical interface capabilities and touch gestures for pagination and similar actions. Both of these interface platforms are expected to change rapidly in the near future.

Ultimately, multimodal interfaces are just one part of the larger movement to establish richer communications interfaces, ones that can expand existing computational functionality and also improve support for human cognition and performance. Figure 18.8 shows this longer-term direction of establishing communications interfaces that are more

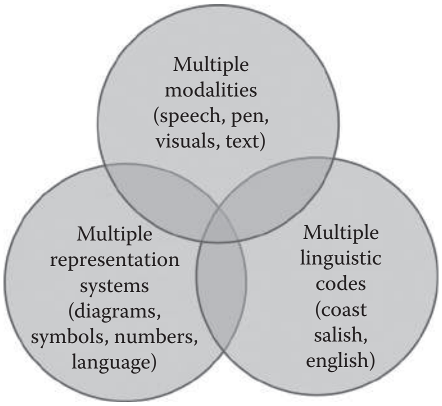

Multiple modalities (speech, pen, visuals, text)
Multiple representation systems (diagrams, symbols, numbers, language)
Multiple linguistic codes (coast salish, english)

FIGURE 18.8 Richly expressive communications interfaces that support multiple modalities, representation systems, and linguistic codes.

capable of supporting people's ability to communicate fluently as they think and work using: (1) multiple modalities (pen, speech, touch, visual, text), (2) multiple representation systems (diagrams, symbols, numbers, language, informal marking), and (3) multiple linguistic codes (western, Asian, indigenous), which may represent one's native language or bilingual/multilingual communication patterns. One major goal of such interfaces is to reduce cognitive load and improve communicative and ideational fluency, as discussed at the end of Section 18.3. Recent results on the cognitive advantages of pen interfaces during real-world problem solving seriously challenge the adequacy of keyboard-centric graphical interfaces, and encourage us to prototype new multimodal systems that incorporate pen input for our education and professional lives (Oviatt in press). An additional goal of developing richer communications interfaces is preservation of the world's diverse heritage languages, many of which are poorly supported by western-European keyboards.

In conclusion, multimodal interfaces are just beginning to model human-like sensory perception and communication patterns. They are recognizing and identifying actions, language, and people that have been seen, heard, or in other ways experienced in the past. They literally reflect and acknowledge the existence of human users, empower them in new ways, and create for them a "voice." They also can be playful and self-reflective interfaces that suggest new forms of human identity as we interact face-to-face with animated personas representing our own kind. In all of these ways novel multimodal interfaces, as primitive as their early bimodal instantiations may be, represent a new multidisciplinary science, a new art form, and a sociopolitical statement about our collective desire to humanize the technology we create.

ACKNOWLEDGMENTS

I thank the National Science Foundation for their support, which enabled me to pursue basic exploratory research on many aspects of multimodal interaction, interface design, and system development. The preparation of this chapter has been supported by NSF Grants IRI-9530666, IIS-0117868, and NSF Special Extension for Creativity (SEC) Grant IIS-9530666, as well as by Incaa Designs nonprofit (www.incaadesigns.org). This work also has been supported by contracts DABT63-95-C-007, N66001-99-D-8503, and NBCHD030010 from DARPA's Information Technology and Information Systems Office, and Grant No. N00014-99-1-0377 from ONR. Any opinions, findings, or conclusions expressed in this chapter are those of the author and do not necessarily reflect the views of the federal agencies sponsoring this work.

I also thank Phil Cohen and others in the Center for Human–Computer Communication for many insightful discussions, and Dana Director and Rachel Coulston for expert assistance with manuscript preparation. Finally, I wish to acknowledge LEA, Inc. for giving permission to reprint Figure 18.7, and to acknowledge ACM for allowing the reprint of Figures 18.3 through 18.6.

REFERENCES

Abry, C., M.-T. Lallouache, and M.-A. Cathiard. 1996. How can coarticulation models account for speech sensitivity to audio-visual desynchronization? In *Speechreading by Humans and Machines: Models, Systems and Applications*, ed. D. G. Stork and M. E. Hennecke, 247–55. New York: Springer Verlag.

Adapx. 2011. http://www.adapx.com/home; retrieved Oct. 5, 2011.

Adjoudani, A., and C. Benoit. 1995. Audio-visual speech recognition compared across two architectures. In *Proceedings of the Eurospeech Conference*, Vol. 2, 1563–66. Madrid, Spain.

Almor, A. 1999. Noun-phrase anaphora and focus: The informational load hypothesis. *Psychol Rev* 106:748–65.

Anoto. 2009. http://www.anoto.com (accessed May 1, 2009).

Argyle, M. 1972. Nonverbal communication in human social interaction. In *Nonverbal Communication*, ed. R. Hinde, 243–67. Cambridge: Cambridge Univ. Press.

Arthur, A., R. Lunsford, M. Wesson, and S. L. Oviatt. 2006. Prototyping novel collaborative multimodal systems: Simulation, data collection and analysis tools for the next decade. In *Eighth International Conference on Multimodal Interfaces (ICMI'06)*, 209–26. New York: ACM.

Baddeley, A. 1992. Working memory. *Science* 255:556–59.

Bangalore, S., and M. Johnston. 2000. Integrating multimodal language processing with speech recognition. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'2000)*, ed. B. Yuan, T. Huang, and X. Tang, Vol. 2, 126–29. Beijing: Chinese Friendship Publishers.

Bangalore, S., and M. Johnston. 2009. Robust understanding in multimodal interfaces. *Comput Ling* 35(3):345–97.

Barthelmess, P., E. Kaiser, X. Huang, and D. Demirdjian. 2005. Distributed pointing for multimodal collaboration over sketched diagrams. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, 10–17. New York: ACM.

Barthelmess, P., and S. Oviatt. 2007. User-centered design for interactive and collaborative multimodal interfaces. In *IEEE Computer Special Issue on Human-Centered Multimedia Interfaces*, Vol. 40(5), ed. N. Sebe, D. Gatica, A. Jaimes, and T. S. Huang. New York: ACM.

Bengio, S. 2004. Multimodal speech processing using asynchronous Hidden Markov Models. *Inf Fusion* 5(2):81–9.

Benoit, C. 2000. The intrinsic bimodality of speech communication and the synthesis of talking faces. In *The Structure of Multimodal Dialogue II*, ed. M. Taylor, F. Neel, and D. Bouwhuis, 485–502. Amsterdam: John Benjamins.

Benoit, C., T. Guiard-Marigny, B. Le Goff, and A. Adjoudani. 1996. Which components of the face do humans and machines best speechread? In *Speechreading by Humans and Machines: Models, Systems, and Applications, Vol. 150 of NATO ASI Series. Series F: Computer and Systems Sciences*, ed. D. G. Stork and M. E. Hennecke, 315–25. Berlin, Germany: Springer-Verlag.

Benoit, C., and B. Le Goff. 1998. Audio-visual speech synthesis from French text: Eight years of models, designs and evaluation at the ICP. *Speech Commun* 26:117–29.

Benoit, C., J.-C. Martin, C. Pelachaud, L. Schomaker, and B. Suhm. 2000. Audio-visual and multimodal speech-based systems. In *Handbook of Multimodal and Spoken Dialogue Systems: Resources, Terminology and Product Evaluation*, ed. D. Gibbon, I. Mertins, and R. Moore, 102–203. Kluwer.

Bernstein, L., and C. Benoit. 1996. For speech perception by humans or machines, three senses are better than one. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'96)*, Vol. 3, 1477–80. New York: IEEE Press.

Bers, J., S. Miller, and J. Makhoul. 1998. Designing conversational interfaces with multimodal interaction. In *DARPA Workshop on Broadcast News Understanding Systems*, 319–21.

Bolt, R. A. 1980. Put-that-there: Voice and gesture at the graphics interface. *Comput Graph* 14(3):262–70.

Brandl, P., C. Forlines, D. Wigdor, M. Haller, and C. Shen. 2008. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal surfaces. In *Conference on Advanced Visual Interfaces*, 154–61.

Bregler, C., and Y. Konig. 1994. Eigenlips for robust speech recognition. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing (IEEE-ICASSP)*, Vol. 2, 669–72. IEEE Press.

Bregler, C., S. Manke, H. Hild, and A. Waibel. 1993. Improving connected letter recognition by lipreading. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (IEEE-ICASSP)*, Vol. 1, 557–60. Minneapolis, MN: IEEE Press.

Brooke, N. M., and E. D. Petajan. 1986. Seeing speech: Investigations into the synthesis and recognition of visible speech movements using automatic image processing and computer graphics. In *Proceedings International Conference Speech Input and Output: Techniques and Applications*, Vol. 258, 104–9.

Calder, J. 1987. Typed unification for natural language processing. In *Categories, Polymorphisms, and Unification*, ed. E. Klein and J. van Benthem, 65–72. Center for Cognitive Science, University of Edinburgh.

Calvert, G., C. Spence, and B. E. Stein, eds. 2004. *The Handbook of Multisensory Processing*. Cambridge, MA: MIT Press.

Carpenter, R. 1990. Typed feature structures: Inheritance, (in)equality, and extensionality. In *Proceedings of the ITK Workshop: Inheritance in Natural Language Processing*, 9–18. Tilburg: Institute for Language Technology and Artificial Intelligence, Tilburg University.

Carpenter, R. 1992. *The Logic of Typed Feature Structures*. Cambridge, UK: Cambridge University Press.

Cassell, J., J. Sullivan, S. Prevost, and E. Churchill, eds. 2000. *Embodied Conversational Agents*. Cambridge, MA: MIT Press.

Chen, L., and M. Harper. 2009. Multimodal floor control shift detection. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, New York: ACM.

Cheyer, A. 1998. MVIEWS: Multimodal tools for the video analyst. In *International Conference on Intelligent User Interfaces (IUI'98)*, 55–62. New York: ACM Press.

Cheyer, A., and L. Julia. 1995. Multimodal maps: An agent-based approach. In *International Conference on Cooperative Multimodal Communication (CMC'95)*, 103–13. Eindhoven, The Netherlands.

Choudhury, T., B. Clarkson, T. Jebara, and S. Pentland. 1999. Multimodal person recognition using unconstrained audio and video. In *Proceedings of the 2nd International Conference on Audio-and-Video-based Biometric Person Authentication*, 176–81. Washington, DC.

Codella, C., R. Jalili, L. Koved, J. Lewis, D. Ling, J. Lipscomb, and D. Rabenhorst et al. 1992. Interactive simulation in a multi-person virtual world. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'92)*, 329–34. New York: ACM Press.

Cohen, P. R., A. Cheyer, M. Wang, and S. C. Baeg. 1994. An open agent architecture. In *AAAI'94 Spring Symposium Series on Software Agents*, 1–8. AAAI Press. (Reprinted in Huhns and Singh (Eds.). 1997. Readings in Agents (pp. 197–204). San Francisco, CA: Morgan Kaufmann.)

Cohen, P. R., M. Dalrymple, D. B. Moran, F. C. N. Pereira, J. W. Sullivan, R. A. Gargan, J. L. Schlossberg, and S. W. Tyler. 1989. Synergistic use of direct manipulation and natural language. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'89)*, 227–34. New York: ACM Press. (Reprinted in Maybury and Wahlster (Eds.). 1998. Readings in Intelligent User Interfaces (pp. 29–37). San Francisco: Morgan Kaufmann.)

Cohen, P. R., M. Johnston, D. McGee, S. Oviatt, J. Pittman, I. Smith, L. Chen, and J. Clow. 1997. Quickset: Multimodal interaction for distributed applications. In *Proceedings of the Fifth ACM International Multimedia Conference*, 31–40. New York: ACM Press.

Cohen, M. M., and D. W. Massaro. 1993. Modeling coarticulation in synthetic visual speech. In *Models and Techniques in Computer Animation*, ed. M. Magnenat-Thalmann and D. Thalmann, 139–56. Tokyo: Springer-Verlag.

Cohen, P. R., and D. McGee. 2004. Tangible multimodal interfaces for safety-critical applications. In *Communications of the ACM*, Vol. 47, 41–6. New York: ACM Press.

Cohen, P. R., D. R. McGee, and J. Clow. 2000. The efficiency of multimodal interaction for a map-based task. In *Proceedings of the Language Technology Joint Conference (ANLP-NAACL 2000)*, 331–8. Seattle, WA: Association for Computational Linguistics Press.

Cohen, P. R., and S. L. Oviatt. 1995. The role of voice input for human-machine communication. *Proc Natl Acad Sci* 92(22):9921–7. Washington, DC: National Academy of Sciences Press.

Cohen, P., C. Swindells, S. Oviatt, and A. Arthur. 2008. A high-performance dual-wizard infrastructure supporting speech and digital pen input. In *Tenth International Conference on Multimodal Interfaces (ICMI'08)*. New York: ACM.

Condon, W. S. 1988. An analysis of behavioral organization. *Sign Lang Stud* 58:55–88.

Dahlbäck, N., A. Jëonsson, and L. Ahrenberg. 1992. Wizard of Oz studies—why and how. In *Proceedings of the International Workshop on Intelligent User Interfaces*, ed. W. D. Gray, W. E. Hefley, and D. Murray, 193–200. New York: ACM Press.

Darves, C., and S. Oviatt. 2004. Talking to digital fish: Designing effective conversational interfaces for educational software.

In *Evaluating Embodied Conversational Agents*, Vol. 7, ed. Z. Ruttkay and C. Pelachaud, 271–92. Dordrecht: Kluwer.

Danninger, M., G. Flaherty, K. Bernardin, H. Ekenel, T. Kohler, R. Malkin, R. Stiefelhagen, and A. Waibel. 2005. The connector-facilitating context-aware communication. In *Proceedings of the International Conference on Multimodal Interfaces*, 69–75. New York: ACM.

Denecke, M., and J. Yang. 2000. Partial information in multimodal dialogue. In *Proceedings of the International Conference on Multimodal Interaction*, 624–33. Beijing, China.

Duncan, L., W. Brown, C. Esposito, H. Holmback, and P. Xue. 1999. *Enhancing Virtual Maintenance Environments with Speech Understanding*. Boeing M&CT TechNet.

Dupont, S., and J. Luettin. 2000. Audio-visual speech modeling for continuous speech recognition. *IEEE Trans Multimedia* 2(3):141–51. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Sept. 2000.

Ekman, P. 1992. Facial expressions of emotion: New findings, new questions. *Am Psychol Soc* 3(1):34–8.

Ekman, P., and W. Friesen. 1978. *Facial Action Coding System*. Palo Alto, CA: Consulting Psychologists Press.

Encarnacao, L. M., and L. Hettinger. 2003. Perceptual multimodal interfaces (special issue). *IEEE Comput Graph Appl*, 54–61.

Epps, J., S. Oviatt, and F. Chen. 2004. Integration of speech and gesture input during multimodal interaction. In *Proceedings of the Australian International Conference on Computer-Human Interaction (OzCHI).*

Ernst, M., and H. Bulthoff. 2004. Merging the sense into a robust whole percept. *Trends Cogn Sci* 8(4):162–9.

Fell, H., H. Delta, R. Peterson, L. Ferrier, Z. Mooraj, and M. Valleau. 1994. Using the baby-babble-blanket for infants with motor problems. In *Proceedings of the Conference on Assistive Technologies (ASSETS'94)*, 77–84. Marina del Rey, CA.

Flanagan, J., and T. Huang. 2003. Multimodal human computer interfaces (special issue). *Proc IEEE'*91(9).

Fridlund, A. 1994. *Human Facial Expression: An Evolutionary View*. New York: Academic Press.

Fukumoto, M., Y. Suenaga, and K. Mase. 1994. Finger-pointer: Pointing interface by image processing. *Comput Graph* 18(5):633–42.

Fuster-Duran, A. 1996. Perception of conflicting audio-visual speech: An examination across spanish and german. In *Speechreading by Humans and Machines: Models, Systems and Applications*, ed. D. G. Stork and M. E. Hennecke, 135–43. New York: Springer Verlag.

Gatica-Perez, D., G. Lathoud, J.-M. Odobez, and I. McCowan. 2005. Multimodal multispeaker probabilistic tracking in meetings. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, 183–190. New York: ACM.

Germesin, S. 2009. Agreement detection in multiparty conversation. In *Proceedings of the Eleventh International Conference on Multimodal Interfaces*. New York: ACM.

Gorniak, P., and D. Roy. 2005. Probabilistic grounding of situated speech using plan recognition and reference resolution. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, 138–43. New York: ACM.

Gupta, A. 2004. Dynamic time windows for multimodal input fusion. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'04).*

Hadar, U., T. J. Steiner, E. C. Grant, and F. Clifford Rose. 1983. Kinematics of head movements accompanying speech during conversation. *Hum Movement Sci* 2:35–46.

Hauptmann, A. G. 1989. Speech and gestures for graphic image manipulation. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'89)*, Vol. 1, 241–5. New York: ACM Press.

Holzman, T. G. 1999. Computer-human interface solutions for emergency medical care. *Interactions* 6(3):13–24.

Hsueh, P., and J. Moore. 2008. Automatic decision detection in meeting speech. In *Machine Learning for Multimodal Interaction, Lecture Notes in Computer Science LNCS 4892*, 168–79. Berlin: Springer.

Huang, X., A. Acero, C. Chelba, L. Deng, D. Duchene, J. Goodman, H. Hon et al. 2000. MiPad: A next-generation PDA prototype. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP 2000)*, Vol. 3, 33–6. Beijing, China: Chinese Military Friendship Publishers.

Huang, X., and S. Oviatt. 2006. Toward adaptive information fusion in multimodal systems. In *Second Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MIML'05)*. Edinburgh, UK: Springer-Verlag.

Iyengar, G., H. Nock, and C. Neti. 2003. Audio-visual synchrony for detection of monologues in video archives. In *Proceedings of ICASSP*.

Jain, A., L. Hong, and Y. Kulkarni. 1999. A multimodal biometric system using fingerprint, face and speech. In *2nd International Conference on Audio- and Video-based Biometric Person Authentication*, 182–7. Washington, DC.

Jain, A., and A. Ross. 2002. Learning user-specific parameters in a multibiometric system. In *Proceedings of the International Conference on Image Processing (ICIP)*. New York: Rochester.

Johnston, M. 2009. Building multimodal applications with EMMA. In *Proceedings of the 11th International Conference on Multimodal Interfaces*, 47–54. New York: ACM.

Johnston, M., P. Baggia, D. Burnett, J. Carter, D. Dahl, G. McCobb, and D. Raggett. 2009. *EMMA: Extensible MultiModal Annotation Markup Language*. http://www.w3.org/TR/2009/REC-emma-20090210. (accessed October 5, 2011).

Johnston, M., P. R. Cohen, D. McGee, S. L. Oviatt, J. A. Pittman, and I. Smith. 1997. Unification-based multimodal integration. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, 281–8. San Francisco, CA: Morgan Kaufmann.

Karshmer, A. I., and M. Blattner, (organizers). 1998. In *Proceedings of the 3rd International ACM Proceedings of the Conference on Assistive Technologies (ASSETS'98)*. New York: ACM.

Kendon, A. 1980. Gesticulation and speech: Two aspects of the process of utterance. In *The Relationship of Verbal and Nonverbal Communication*, ed. M. Key, 207–27. The Hague: Mouton.

Kobsa, A., J. Allgayer, C. Reddig, N. Reithinger, D. Schmauks, K. Harbusch, and W. Wahlster. 1986. Combining deictic gestures and natural language for referent identification. In *Proceedings of the 11th International Conference on Computational Linguistics*, 356–61. Bonn, Germany.

Koons, D., C. Sparrell, and K. Thorisson. 1993. Integrating simultaneous input from speech, gaze, and hand gestures. In *Intelligent Multimedia Interfaces*, ed. M. Maybury, 257–76. Cambridge, MA: MIT Press.

Kopp, S., P. Tepper, and J. Cassell. 2004. Towards integrated microplanning of language and iconic gesture for multimodal output. In *Proceedings of the 6th International Conference on Multimodal Interfaces*, 97–104. New York: ACM.

Kricos, P. B. 1996. Differences in visual intelligibility across talkers. In *Speechreading by Humans and Machines: Models, Systems and Applications*, ed. D. G. Stork and M. E. Hennecke, 43–53. New York: Springer Verlag.

Kumar, S., and P. R. Cohen. 2000. Towards a fault-tolerant multi-agent system architecture. In *Fourth International Conference on Autonomous Agents 2000*, 459–66. Barcelona, Spain: ACM Press.

Lalanne, D., F. Evequoz, M. Rigamonti, B. Dumas, and R. Ingold. 2008. An ego-centric and tangible approach to meeting indexing and browsing. In *Machine Learning and Multimodal Interaction*, ed. A. Popescu-Belis, S. Renals, and H. Bourlard, Lecture Notes in Computer Science LNCS 4892, 84–95. Berlin: Springer.

Lalanne, D., A. Lisowska, E. Bruno, M. Flynn, M. Georgescul, M. Guillemot, B. Janvier et al. 2005. The IM2 meeting browser family, Technical Report, Friborg.

Leatherby, J. H., and R. Pausch. 1992. Voice input as a replacement for keyboard accelerators in a mouse-based graphical editor: An empirical study. *J Am Voice Input Output Soc* 11(2).

Leitner, J., J. Powell, P. Brandl, T. Seifried, M. Haller, B. Dorsay, and P. To. 2009. FLUX: A tilting multi-touch and pen-based surface. In *Proceedings of Computer-Human Interaction Conference*, 11–16. New York: ACM Press.

Lisowska, A. 2007. *Multimodal Interface Design for Multimedia Meeting Content Retrieval*, PhD thesis, Switzerland: University of Geneva.

Liverscribe. www.livescribe.com; retrieved Oct. 5, 2011.

Liwicki, M., and S. El-Neklawy. 2009. *Enhancing a Multi-Touch Table with Write Functionality*. Kyoto, Japan: Workshop on MPR.

Lunsford, R., S. L. Oviatt, and A. Arthur. 2006. Toward open-microphone engagement for multiparty interactions. In *Eighth International Conference on Multimodal Interfaces (ICMI'06)*, 273–80. New York: ACM.

Lunsford, R., S. Oviatt, and R. Coulston. 2005. Audio-visual cues distinguishing self- from system-directed speech in younger and older adults. In *Seventh International Conference on Multimodal Interfaces (ICMI'05)*, 167–74. New York: ACM.

Martin, D. L., A. J. Cheyer, and D. B. Moran. 1999. The open agent architecture: A framework for building distributed software systems. *Appl Artif Intell* 13:91–128.

Massaro, D. W. 1996. Bimodal speech perception: A progress report. In *Speechreading by Humans and Machines: Models, Systems and Applications*, ed. D. G. Stork and M. E. Hennecke, 79–101. New York: Springer Verlag.

Massaro, D. W., and M. M. Cohen. 1990. Perception of synthesized audible and visible speech. *Psychol Sci* 1(1):55–63.

Massaro, D. W., and D. G. Stork. 1998. Sensory integration and speechreading by humans and machines. *Am Scientist* 86:236–44.

McCowan, I., D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang. 2005. Automatic analysis of multimodal group actions in meetings. *IEEE Trans Pattern Anal Mach Intell (PAMI)* 27(3):305–17.

McGee, D. 2003. *Augmenting Environments with Multimodal Interaction*, Oregon Health & Science University, Doctoral dissertation.

McGrath, M., and Q. Summerfield. 1985. Intermodal timing relations and audio-visual speech recognition by normal-hearing adults. *J Acoust Soc Am* 77(2):678–85.

McGurk, H., and J. MacDonald. 1976. Hearing lips and seeing voices. *Nature* 264:746–8.

McLeod, A., and Q. Summerfield. 1987. Quantifying the contribution of vision to speech perception in noise. *Br J Audiol* 21:131–41.

McNeill, D. 1992. *Hand and Mind: What Gestures Reveal about Thought*. Chicago, IL: University of Chicago Press.

Meier, U., W. Hürst, and P. Duchnowski. 1996. Adaptive bimodal sensor fusion for automatic speechreading. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (IEEE-ICASSP)*, 833–6. New York: IEEE Press.

Morency, L.-P., C. Sidner, C. Lee, and T. Darrell. 2005. Contextual recognition of head gestures. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, 18–24. New York: ACM.

Morimoto, C., D. Koons, A. Amir, M. Flickner, and S. Zhai. 1999. Keeping an eye for HCI. In *Proceedings of SIBGRAPI'99, XII Brazilian Symposium on Computer Graphics and Image Processing*, 171–6.

Mousavi, S. Y., R. Low, and J. Sweller. 1995. Reducing cognitive load by mixing auditory and visual presentation modes. *J Educ Psychol* 87(2):319–34.

Naughton, K. 1996. Spontaneous gesture and sign: A study of ASL signs co-occurring with speech. In *Proceedings of the Workshop on the Integration of Gesture in Language & Speech*, ed. L. Messing, 125–34. Univ. of Delaware.

Neal, J. G., and S. C. Shapiro. 1991. Intelligent multimedia interface technology. In *Intelligent User Interfaces*, ed. J. Sullivan and S. Tyler, 11–43. New York: ACM Press.

Negroponte, N. 1978. *The Media Room*. Report for ONR and DARPA. Cambridge, MA: MIT, Architecture Machine Group.

Neti, C., G. Iyengar, G. Potamianos, and A. Senior. 2000. Perceptual interfaces for information interaction: Joint processing of audio and visual information for human-computer interaction. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'2000)*, Vol. 3, ed. B. Yuan, T. Huang, and X. Tang, 11–4. Beijing, China: Chinese Friendship Publishers.

Oliver, N., and E. Horvitz. 2005. S-SEER: Selective perception in a multimodal office activity system. *Int J Comput Vision Image Understanding*, 198–224.

Oviatt, S. L. 1997. Multimodal interactive maps: Designing for human performance. *Hum Comput Interact [Special issue on Multimodal Interfaces]* 12:93–129.

Oviatt, S. L. 1999a. Mutual disambiguation of recognition errors in a multimodal architecture. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'99)*, 576–83. New York: ACM Press.

Oviatt, S. L. 1999b. Ten myths of multimodal interaction. *Commun ACM* 42(11):74–81. New York: ACM Press. (Translated into Chinese by Jing Qin and published in the Chinese journal Computer Application.)

Oviatt, S. L. 2000a. Multimodal system processing in mobile environments. In *Proceedings of the Thirteenth Annual ACM Symposium on User Interface Software Technology (UIST'2000)*, 21–30. New York: ACM Press.

Oviatt, S. L. 2000b. Taming recognition errors with a multimodal architecture. *Commun ACM* 43(9):45–51. New York: ACM Press.

Oviatt, S. L. 2000c. Multimodal signal processing in naturalistic noisy environments. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'2000)*, Vol. 2, ed. B. Yuan, T. Huang, and X. Tang, 696–9. Beijing, China: Chinese Friendship Publishers.

Oviatt, S. L. 2002. Breaking the robustness barrier: Recent progress in the design of robust multimodal systems. In *Advances in Computers*, vol. 56, ed. M. Zelkowitz, Academic Press, 305–41.

Oviatt, S. L. 2003. Advances in robust multimodal interfaces. *IEEE Comput Graphics Appl* 62–8. (special issue on Perceptual Multimodal Interfaces).

Oviatt, S. L. 2006. Human-centered design meets cognitive load theory: Designing interfaces that help people think. In *Proceedings of the ACM Conference on Multimedia, Special Session on "Human-Centered Multimedia Systems,"* 871–80. New York: ACM.

Oviatt, S., A. Arthur, Y. Brock, and J. Cohen. 2007. Expressive pen-based interfaces for math education. In *Proceedings of the Conference on Computer-Supported Collaborative Learning*. International Society of the Learning Sciences. In C. Chinn, G. Erkens and S. Puntambekar, eds., Proceedings of the Conference on Computer Supported Collaborative Learning 2007: Of Mice, Minds & Society, *International Society of the Learning Sciences* 8(2):569–578.

Oviatt, S., A. Arthur, and J. Cohen. 2006. Quiet interfaces that help students think. In *Proceedings of the Conference on User Interface Software Technology*, 191–200. New York, ACM Press.

Oviatt, S. L., J. Bernard, and G. Levow. 1999. Linguistic adaptation during error resolution with spoken and multimodal systems. *Lang Speech* 41(3–4):415–38 (special issue on "Prosody and Speech").

Oviatt, S. L., and A. Cohen. 2010a. Supporting students' thinking marks: Designing accessible interfaces for science education. *American Educational Research Association Conference*.

Oviatt, S. L., and A. Cohen. 2010b. Toward high-performance communication interfaces for science problem solving. *Journal of Science Education and Technology* 19(6):515–531.

Oviatt, S. L. (in press). *The Future of Educational Interfaces*, Routledge Press, forthcoming in 2012.

Oviatt, S. L., and P. R. Cohen. 1991. Discourse structure and performance efficiency in interactive and noninteractive spoken modalities. *Comput Speech Lang* 5(4):297–326.

Oviatt, S. L., and P. R. Cohen. 2000. Multimodal systems that process what comes naturally. *Commun ACM* 43(3):45–53. New York: ACM Press.

Oviatt, S. L., P. R. Cohen, M. W. Fong, and M. P. Frank. 1992. A rapid semi-automatic simulation technique for investigating interactive speech and handwriting. In *Proceedings of the International Conference on Spoken Language Processing, 2*, 1351–54. Univ. of Alberta.

Oviatt, S. L., P. R. Cohen, and M. Q. Wang. 1994. Toward interface design for human language technology: Modality and structure as determinants of linguistic complexity. *Speech Commun* 15:283–300. European Speech Communication Association.

Oviatt, S. L., P. R. Cohen, L. Wu, J. Vergo, L. Duncan, B. Suhm, J. Bers et al. 2000. Designing the user interface for multimodal speech and gesture applications: State-of-the-art systems and research directions. *Hum Comput Interact* 15(4):263–322. (to be reprinted in J. Carroll (Ed.) Human-Computer Interaction in the New Millennium, Addison-Wesley Press: Boston, 2001).

Oviatt, S. L., R. Coulston, and R. Lunsford. 2004. When do we interact multimodally? Cognitive load and multimodal communication patterns. In *Proceedings of the Sixth International Conference on Multimodal Interfaces (ICMI'04)*, 129–36. New York: ACM Press.

Oviatt, S. L., R. Coulston, S. Shriver, B. Xiao, R. Wesson, R. Lunsford, and L. Carmichael. 2003. Toward a theory of organized multimodal integration patterns during human-computer interaction. In *Proceedings of the International Conference on Multimodal Interfaces (ICMI'03)*, 44–51. New York: ACM Press.

Oviatt, S. L., A. DeAngeli, and K. Kuhn. 1997. Integration and synchronization of input modes during multimodal human-computer interaction. In *Proceedings of Conference on Human Factors in Computing Systems (CHI'97)*, 415–22. New York: ACM Press.

Oviatt, S. L., M. Flickner, and T. Darrell, eds. 2004. Multimodal interfaces that flex, adapt and persist. *Commun ACM* (special issue) 47(1).

Oviatt, S. L., and K. Kuhn. 1998. Referential features and linguistic indirection in multimodal language. In *Proceedings of the International Conference on Spoken Language Processing, 6*, 2339–42. Syndey, Australia: ASSTA, Inc.

Oviatt, S. L., and R. Lunsford. 2005. Multimodal interfaces for cell phones and mobile technology. *Int J Speech Technol* 8(2):127–32.

Oviatt, S. L., R. Lunsford, and R. Coulston. 2005. Individual differences in multimodal integration patterns: What are they and why do they exist? In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'05), CHI Letters*, 241–9. New York: ACM Press.

Oviatt, S. L., and E. Olsen. 1994. Integration themes in multimodal human-computer interaction. In *Proceedings of the International Conference on Spoken Language Processing, 2*, ed. K. Shirai, S. Furui, and K. Kakehi, 551–4. Acoustical Society of Japan.

Oviatt, S. L., C. Swindells, and A. Arthur. 2008. Implicit user-adaptive system engagement in speech and pen interfaces. In *Conference on Human Factors in Computing Systems (CHI '08), CHI Letters*, 969–78. New York: ACM.

Oviatt, S. L., and R. van Gent. 1996. Error resolution during multimodal human-computer interaction. In *Proceedings of the International Conference on Spoken Language Processing, 2*, 204–7. University of Delaware Press.

Pankanti, S., R. M. Bolle, and A. Jain, eds. 2000. Biometrics: The future of identification. *Computer* 33(2):46–80.

Pavlovic, V., G. Berry, and T. S. Huang. 1997. Integration of audio/visual information for use in human-computer intelligent interaction. In *Proceedings of IEEE International Conference on Image Processing*, 121–4. IEEE Press.

Pavlovic, V., and T. S. Huang. 1998. Multimodal prediction and classification on audio-visual features. In *AAAI'98 Workshop on Representations for Multi-modal Human-Computer Interaction*, 55–59. Menlo Park, CA: AAAI Press.

Pavlovic, V., R. Sharma, and T. Huang. 1997. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans Pattern Anal Mach Intell* 19(7):677–95.

Pentland, S. 2005. Socially aware computation and communication. *IEEE Comput* 63–70.

Popescu-Belis, A., P. Baudrion, M. Flynn, and P. Wellner. 2008. Towards an objective test for meeting browsers: The BET4TQB pilot experiment. In *Machine Learning and Multimodal Interaction*, Lecture Notes in Computer Science LNCS 4892, 108–19. Berlin: Springer.

Popescu-Belis, A., S. Renals, and H. Bourlard, eds. 2008. Machine learning for multimodal interaction. In *Fourth International Workshop on MLMI'07*, Lecture Notes in Computer Science, Berlin: Springer Publ.

Popescu-Belis, A., and M. Georgescul. 2006. TQB: Accessing multimedia data using a transcript-based query and browsing interface. In *Proceedings of LREC*, 1560–65.

Potamianos, G., and C. Neti. 2001. Automatic speechreading of impaired speech. In *Proceedings of the International Conference on Auditory-Visual Speech Processing*, Aalborg, Denmark, 177–82.

Potamianos, G., C. Neti, G. Gravier, and A. Garg. 2003. Automatic recognition of audio-visual speech: Recent progress and challenges. *Proc IEEE* 91(9):1–18.

Petajan, E. D. 1984. *Automatic Lipreading to Enhance Speech Recognition*, PhD thesis, University of Illinois at Urbana-Champaign.

Pfleger, N. 2004. Context-based multimodal fusion. In *Proceedings of the 6th International Conference on Multimodal Interfaces*, 265–72. New York: ACM.

Poddar, I., Y. Sethi, E. Ozyildiz, and R. Sharma. 1998. Toward natural gesture/speech HCI: A case study of weather narration. In *Proceedings 1998 Workshop on Perceptual User Interfaces (PUI'98)*, ed. M. Turk, 1–6. San Francisco, CA.

Reithinger, N., J. Alexandersson, T. Becker, A. Blocher, R. Engel, M. Lockelt, J. Muller et al. 2003. Multimodal architectures and frameworks: SmartKom: Adaptive and flexible multimodal access to multiple applications. In *Proceedings of the 5th International Conference on Multimodal Interfaces*, 101–8. New York: ACM.

Robert-Ribes, J., J.-L. Schwartz, T. Lallouache, and P. Escudier. 1998. Complementarity and synergy in bimodal speech: Auditory, visual, and auditory-visual identification of French oral vowels in noise. *J Acoust Soc Am* 103(6):3677–89.

Rogozan, A., and P. Deglise. 1998. Adaptive fusion of acoustic and visual sources for automatic speech recognition. *Speech Commun* 26(1–2):149–61.

Rubin, P., E. Vatikiotis-Bateson, and C. Benoit, eds. 1998. Audio-visual speech processing [Special issue]. *Speech Commun* 26:1–2.

Rudnicky, A., and A. Hauptman. 1992. Multimodal interactions in speech systems. In *Multimedia Interface Design*, ed. M. Blattner and R. Dannenberg, 147–72. New York: ACM Press.

Ruiz, N., F. Chen, and S. Oviatt. 2010. Multimodal human-computer and human-human interaction. In *Multimodal Signal Processing: Theory and Applications for Human-Computer Interaction,* ed. J.-P. Thiran, F. Marques and H. Bourlard, 229–56. Amsterdam: Elsevier.

Sekiyama, K., and Y. Tohkura. 1991. McGurk effect in non-English listeners: Few visual effects for Japanese subjects hearing Japanese syllables of high auditory intelligibility. *J Acoust Soc Am* 90:1797–805.

Seneff, S., D. Goddeau, C. Pao, and J. Polifroni. 1996. Multimodal discourse modelling in a multi-user multi-domain environment. In *Proceedings of the International Conference on Spoken Language Processing*, Vol. 1, ed. T. Bunnell and W. Idsardi, 192–5. University of Delaware & A.I. duPont Institute.

Shaikh, A., S. Juth, A. Medl, I. Marsic, C. Kulikowski, and J. Flanagan. 1997. An architecture for multimodal information fusion. In *Proceedings of the Workshop on Perceptual User Interfaces (PUI'97)*, 91–3. Banff, Canada.

Sharma, R., T. S. Huang, V. I. Pavlovic, K. Schulten, A. Dalke, J. Phillips, M. Zeller, W. Humphrey, Y. Zhao, Z. Lo, and S. Chu. 1996. Speech/gesture interface to a visual computing environment for molecular biologists. In *Proceedings of 13th International Conference on Pattern Recognition (ICPR'96)*, Vol. 3, 964–68.

Sharma, R., V. I. Pavlovic, and T. S. Huang. 1998. Toward multimodal human-computer interface. *Proc IEEE* 86(5) [Special issue on Multimedia Signal Processing]:853–60.

Silsbee, P. L., and Q. Su. 1996. Audiovisual sensory intergration using Hidden Markov Models. In *Speechreading by Humans and Machines: Models, Systems and Applications*, ed. D. G. Stork and M. E. Hennecke, 489–504. New York: Springer Verlag.

Siroux, J., M. Guyomard, F. Multon, and C. Remondeau. 1995. Modeling and processing of the oral and tactile activities in the georal tactile system. In *International Conference on Cooperative Multimodal Communication, Theory & Applications*. Eindhoven, Netherlands.

Stork, D. G., and M. E. Hennecke, eds. 1995. *Speechreading by Humans and Machines*. New York: Springer Verlag.

Suhm, B. 1998. *Multimodal Interactive Error Recovery for Non-Conversational Speech User Interfaces*. Ph.D. thesis, Germany: Shaker Verlag: Fredericiana University.

Sumby, W. H., and I. Pollack. 1954. Visual contribution to speech intelligibility in noise. *J Acoust Soc Am* 26:212–15.

Summerfield, A. Q. 1992. Lipreading and audio-visual speech perception. *Philos Trans R Soc London Ser B* 335:71–8.

Sweller, J. 1988. Cognitive load during problem solving: Effects on learning. *Cognitive Sci* 12:257–85.

Tang, A., P. McLachlan, K. Lowe, C. Saka, and K. MacLean. 2005. Perceiving ordinal data haptically under workload. In *Proceedings of the Seventh International Conference on Multimodal Interfaces*, 317–24. New York: ACM.

Tomlinson, M. J., M. J. Russell, and N. M. Brooke. 1996. Integrating audio and visual information to provide highly robust speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (IEEE-ICASSP)*, Vol. 2, 821–4. IEEE Press.

Turk, M., and G. Robertson, eds. 2000. Perceptual user interfaces [Special issue]. *Commun ACM* 43(3):32–70.

Vatikiotis-Bateson, E., K. G. Munhall, M. Hirayama, Y. V. Lee, and D. Terzopoulos. 1996. The dynamics of audiovisual behavior of speech. In *Speechreading by Humans and Machines: Models, Systems, and Applications, Vol. 150 of NATO ASI Series. Series F: Computer and Systems Sciences*, ed. D. G. Stork and M. E. Hennecke, 221–32. Berlin, Germany: Springer-Verlag.

Vo, M. T., R. Houghton, J. Yang, U. Bub, U. Meier, A. Waibel, and P. Duchnowski. 1995. Multimodal learning interfaces. In *Proceedings of the DARPA Spoken Language Technology* Workshop.

Vo, M. T., and C. Wood. 1996. Building an application framework for speech and pen input integration in multimodal learning interfaces. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing (IEEE-ICASSP)*, Vol. 6, 3545–48. IEEE Press.

Wahlster, W. 1991. User and discourse models for multimodal communciation. In *Intelligent User Interfaces*, Chap. 3, ed. J. W. Sullivan and S. W. Tyler, 45–67. New York: ACM Press.

Wahlster, W. 2001. SmartKom: multimodal dialogs with mobile web users. In *Proceedings of the Cyber Assist International Symposium*, 33–4. Tokyo International Forum.

Waibel, A., B. Suhm, M. T. Vo, and J. Yang. 1997. Multimodal interfaces for multimedia information agents. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing (IEEE-ICASSP)*, Vol. 1, 167–70. IEEE Press.

Wang, J. 1995. Integration of eye-gaze, voice and manual response in multimodal user interfaces. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, 3938–42. IEEE Press.

Whittaker, S., R. Laban, and S. Tucker. 2005. *Analysing Meeting records: An Ethnographic Study and Technical Implications*, in Lecture Notes in Computer Science 3869, Machine Learning for Multimodal Interaction, New York: Springer.

Wickens, C. D., D. L. Sandry, and M. Vidulich. 1983. Compatibility and resource competition between modalities of input, central processing, and output. *Hum Factors* 25:227–48.

Wu, L., S. Oviatt, and P. Cohen. 1999. Multimodal integration—A statistical view. *IEEE Trans Multimedia* 1(4):334–41.

Xiao, B., C. Girand, and S. Oviatt. 2002. Multimodal integration patterns in children. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP'02)*, ed. J. Hansen and B. Pellom, 629–32. Denver, CO: Casual Prod. Ltd.

Xiao, B., R. Lunsford, R. Coulston, R. Wesson, and S. L. Oviatt. 2003. Modeling multimodal integration patterns and performance in seniors: Toward adaptive processing of individual differences. In *Proceedings of the International Conference on Multimodal Interfaces (ICMI'03)*, New York: ACM Press. 265–72.

Zhai, S., C. Morimoto, and S. Ihde. 1999. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI'99)*, 246–53. New York: ACM Press.

19 Systems That Adapt to Their Users

Anthony Jameson and Krzysztof Z. Gajos

CONTENTS

19.1 INTRODUCTION

This chapter covers a broad range of interactive systems which have one idea in common: that it can be worthwhile for a system to learn something about individual users and adapt its behavior to them in some nontrivial way.

A representative example is shown in Figure 19.1: the CommunityCommands recommender plug-in for AutoCAD (introduced by Matejka et al. [2009] and discussed more extensively by Li et al. [2011]). To help users deal with the hundreds of commands that AutoCAD offers—of which most users know only a few dozen—CommunityCommands

(1) gives the user easy access to several recently used commands, which the user may want to invoke again soon; and (2) more proactively suggests commands that this user has not yet used but may find useful, given the type of work they have been doing recently.

19.1.1 Concepts

A key idea embodied in CommunityCommands and the other systems discussed in this chapter is that of *adaptation to the individual user*. Depending on their function and form, particular types of systems that adapt to their users have been given labels including *adaptive user interfaces*, *software agents*, *recommender systems*, and *personalization*. To be able to discuss the common issues that all of these systems raise, we will refer to them with a term that describes their common property explicitly: *user-adaptive systems*. Figure 19.2 introduces some concepts that can be applied to any user-adaptive system; Figure 19.3 shows the form that they take in recommendations generated by CommunityCommands.

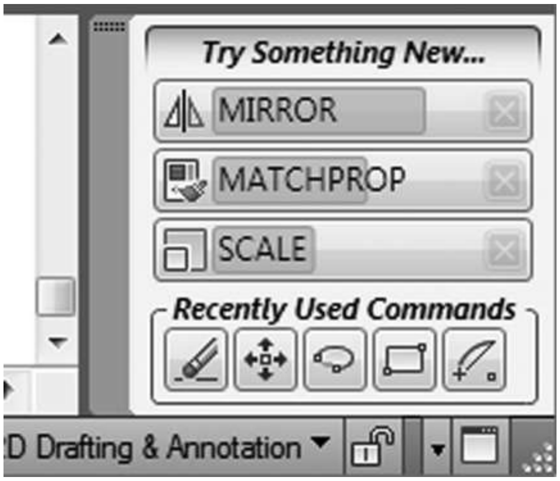


FIGURE 19.1 Screenshot showing how CommunityCommands recommends commands to a user. (Image supplied by Justin Matejka. The length of the darker bar for a command reflects its estimated relevance to the user's activities. When the user hovers over a command in this interface, a tooltip appears that explains the command and might show usage hints provided by colleagues. See also http://www.autodesk.com/research.)

A user-adaptive system makes use of some type of information about the current individual user, such as the commands that the user has executed. In the process of *user model acquisition*, the system performs some type of learning and/or inference on the basis of the information about the user to arrive at some sort of *user model*, which in general concerns only limited aspects of the user (such as their pattern of command use). In the process of *user model application*, the system applies the user model to the relevant features of the current situation to determine how to adapt its behavior to the user; this process may be straightforward, or it can involve some fairly sophisticated decision making on the part of the system.

A user-adaptive system can be defined as an interactive system that adapts its behavior to individual users on the basis of processes of user model acquisition and application that involve some form of learning, inference, or decision making.

The second half of the definition is necessary because otherwise any interactive system could be said to "adapt" to its users, even if it just responds straightforwardly to key presses. It is the processes of user model acquisition and application that raise many common issues and challenges that characterize user-adaptive systems.

This definition also distinguishes user-adaptive systems from purely *adaptable* systems: ones that offer the user an opportunity to configure or otherwise influence the system's longer term behavior (e.g., by choosing options that determine the appearance of the user interface [UI]). Often, what works best is a carefully chosen combination of adaptation and adaptability. For example, if the user of CommunityCommands is not interested in the command MATCHPROP, they can click on the "close" button next to it to specify that it should not be recommended again. Keeping

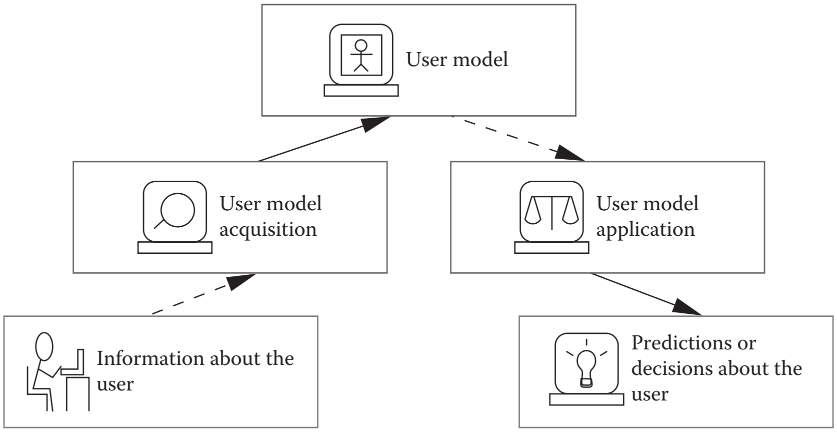

User model
User model acquisition
User model application
Information about the user
Predictions or decisions about the user

FIGURE 19.2 General schema for the processing in a user-adaptive system. (Dotted arrows: use of information; solid arrows: production of results.)

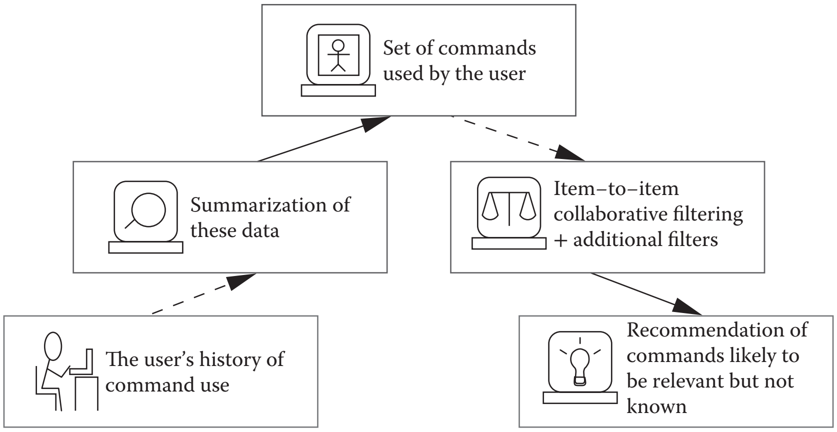

Set of commands used by the user
Summarization of these data
Item–to–item collaborative filtering + additional filters
The user's history of command use
Recommendation of commands likely to be relevant but not known

FIGURE 19.3 Overview of adaptation in CommunityCommands.

the user "in the loop" in this way can be an essential part of effective and well-accepted adaptation.

19.1.2 Chapter Preview

Sections 19.2 and 19.3 address the question "What can user-adaptivity be good for?" They examine in turn a number of different functions that can be served by user adaptivity, giving examples ranging from familiar commercially deployed systems to research prototypes. Section 19.4 discusses some usability challenges that are especially important in connection with user-adaptive systems, challenges which stimulated much of the controversy that surrounded these systems when they first began to appear in the 1980s and 1990s. Section 19.5 considers a key design decision: what types of information about each user should be collected? The chapter concludes with a reflection on the current state of the art and the future challenges for user-adaptive systems.*

19.2 FUNCTIONS: SUPPORTING SYSTEM USE

Some of the ways in which user adaptivity can be helpful involve support for a user's efforts to operate a system successfully and effectively. This section considers four types of support.

19.2.1 Adaptively Offering Help

The first form is the one illustrated by the CommunityCommands recommender: in cases where it is not sufficiently obvious to users how they should operate a given application, a help system can adaptively offer information and advice about how to use it—and perhaps also execute some actions on behalf of the user. That is, the system can act like a helpful friend who is looking over the user's shoulder—a service which users often greatly appreciate but which is not in general easy to automate effectively. The adaptation can make the help that is offered more relevant to the user's needs than the more commonly encountered user-independent help.

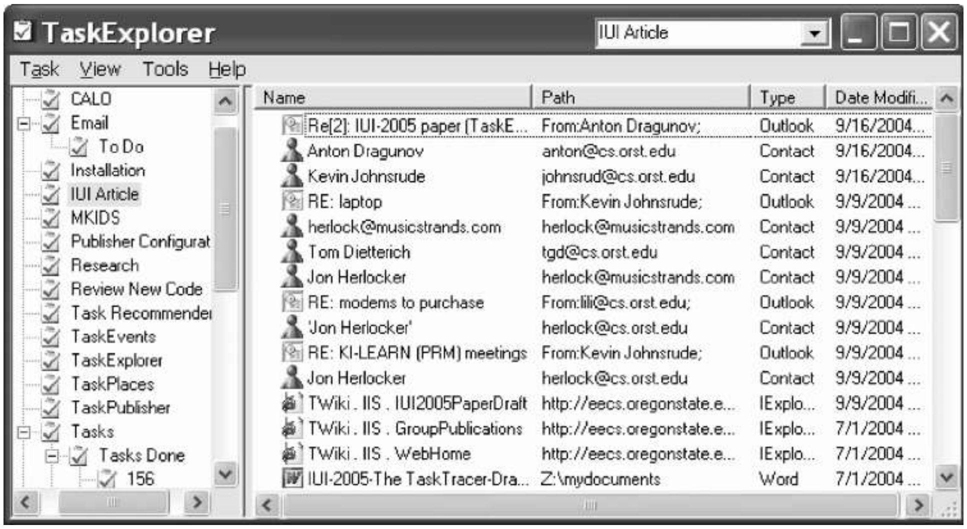
The main way in which CommunityCommands helps the user is by recommending possibly useful commands that the user has not yet used. The basic recommendation technique is *collaborative filtering*, which is discussed later in this chapter in connection with systems that recommend products. The central idea is: "People who use commands like the ones that you have been using also use the following commands, which you may not be familiar with: . . . ."

Matejka et al. (2009) explain how the basic collaborative filtering algorithm had to be adapted and supplemented to yield good performance for command recommendation. For example, if a user already knows the command *A*, it makes little sense to recommend a command *B*, which is just a similar or less efficient way of achieving the same effect; so handcrafted rules were added to prevent such recommendations.

Experience with the deployment of CommunityCommands as an AutoCAD plug-in has indicated that this approach appears to have general feasibility and usefulness for systems that offer a large number of commands. This case study can also be seen as a successful application of the strategy of looking for a relatively lightweight approach to adaptation that still offers considerable added value. Attention to the details of the adaptive algorithms and of the UI design appears to be more important here than the use of more complex adaptation technology.

Systems that offer help in an adaptive way have a long history. Perhaps the most obvious—but also the most difficult—scenario is one in which a user is trying to achieve a particular goal (e.g., align the objects in a drawing in a particular way) but does not know how to achieve the goal with the system. A helper could in principle automatically recognize the user's difficulty and suggest a way of solving the problem. A good deal of research into the development of systems that can take the role of a knowledgeable helper was conducted in the

* Interested readers may also want to consult the chapters on this topic in the first two editions of this handbook (Jameson 2003, 2008), which include discussions of earlier user-adaptive systems that can still serve as instructive examples, as well as discussions of typical issues and methods associated with empirical studies of user-adaptive systems.



Name	Path	Type	Date Modifi...
Re[2]: IUI-2005 paper (TaskE...	From:Anton Dragunov;	Outlook	9/16/2004...
Anton Dragunov	anton@cs.orst.edu	Contact	9/16/2004...
Kevin Johnsrude	johnsrud@cs.orst.edu	Contact	9/16/2004...
RE: laptop	From:Kevin Johnsrude;	Outlook	9/9/2004 ...
herlock@musicstrands.com	herlock@musicstrands.com	Contact	9/9/2004 ...
Tom Dietterich	tgd@cs.orst.edu	Contact	9/9/2004 ...
Jon Herlocker	herlock@musicstrands.com	Contact	9/9/2004 ...
RE: modems to purchase	From:lili@cs.orst.edu;	Outlook	9/9/2004 ...
'Jon Herlocker'	herlock@cs.orst.edu	Contact	9/9/2004 ...
RE: KI-LEARN (PRM) meetings	From:Kevin Johnsrude;	Outlook	9/9/2004 ...
Jon Herlocker	herlock@cs.orst.edu	Contact	9/9/2004 ...
TWiki . IIS . IUI2005PaperDraft	http://eecs.oregonstate.e...	IExplo...	9/9/2004 ...
TWiki . IIS . GroupPublications	http://eecs.oregonstate.e...	IExplo...	7/1/2004 ...
TWiki . IIS . WebHome	http://eecs.oregonstate.e...	IExplo...	7/1/2004 ...
IUI-2005-The TaskTracer-Dra...	Z:\mydocuments	Word	7/1/2004 ...

FIGURE 19.4 Screenshot showing one of the ways in which TaskTracer helps a user to find resources associated with a given *project*. Here, the various resources associated with "IUI Article" are listed in order of recency. (Image retrieved from http://eecs.oregonstate.edu/TaskTracer/ in March 2011, reproduced with permission of Thomas Dietterich.)

1980s, especially in connection with the complex operating system Unix.* During the 1990s, such work became less frequent, perhaps partly because of a recognition of the fundamental difficulties involved: it is in general hard to recognize what goal a user is pursuing when the user is not performing actions that serve their goal. And spontaneously offering help can be distracting, since the system cannot be sure that the user is interested in getting help. The Office Assistant, an ambitious attempt at adaptive help introduced in Microsoft Office 97, was given a mixed reception, partly because of the inherent difficulty of its task but especially because of its widely perceived obtrusiveness (cf. Section 19.4).

For these reasons, more recent research has focused on less ambitious but still potentially useful ways of adaptively offering help. A strategy in this category—one which is quite different from that of CommunityCommands—is to view the process of offering help as involving collaboration and dialog with the user. A representative of this paradigm is the DiamondHelp system, which assists users in the operation of complex consumer devices (see, e.g., Rich et al. [2005]). DiamondHelp is somewhat reminiscent of the (nonadaptive) "wizards" that walk users through procedures such as the configuration of new software; but is more flexible and adaptive in that it applies a *mixed-initiative* paradigm, allowing the user to perform sequences of actions on their own if they like and trying to keep track of what they are doing. Rich (2009) offers a recent discussion of this general paradigm.

19.2.2 Taking Over Parts of Routine Tasks

Another function of adaptation involves taking over some of the work that the user would normally have to perform herself—routine tasks that may place heavy demands on a user's time, though typically not on their intelligence or knowledge. Two traditionally popular candidates for automation of this sort (discussed briefly below) have been the sorting of e-mail and the scheduling of appointments and meetings.

The system TaskTracer (Figure 19.4) illustrates a number of typical functionalities of systems in this category.† The tedious work that is taken over by TaskTracer is not a single, separate chore but rather parts of many of the routine subtasks that are involved in everyday work with a normal desktop (or laptop) computer. The central insight is that a user is typically multitasking among a set of *projects*, each of which is associated with a diverse set of *resources*, such as files, web pages, and e-mail messages. Since these resources tend to be stored in different places and used by different applications, a significant proportion of everyday computer work involves locating and accessing the resources that are relevant to the project that is currently in the focus of attention.

The user of TaskTracer creates a structured list of projects that they sometimes work on; once they have done so, the system does two things largely autonomously: (1) By observing the user, it learns which resources are associated with which projects; (2) It tries to figure out which project the user is working on at any given moment (see, e.g., Shen et al. [2009]). As can be seen in Figure 19.5, these two functions constitute the adaptive aspects of the system.

Even if these inferences by the system are not entirely accurate, they can help the user in various ways: for example, when the user wants to save a document that they have created, TaskTracer can save them some mouse clicks by suggesting two or three folders associated with the current project in which they might like to store the new file. And

* A collection of papers from this period appeared in a volume edited by Hegner et al. (2001).

† See Dietterich et al. (2010), for a recent comprehensive discussion of TaskTracer and http://eecs.oregonstate.edu/TaskTracer/ for further information and references.

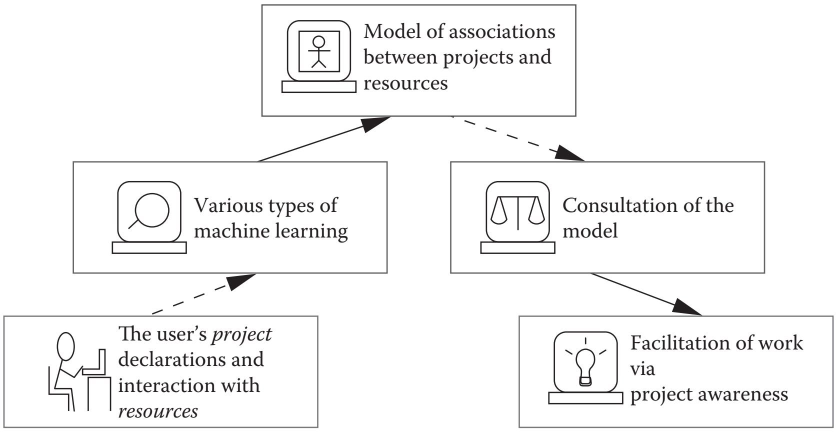

Model of associations between projects and resources
Various types of machine learning
Consultation of the model
The user's *project* declarations and interaction with *resources*
Facilitation of work via project awareness

FIGURE 19.5 Overview of adaptation in TaskTracer.

when a user switches to a project, TaskTracer can offer a list of the resources associated with the current project, sorted by recency of access, so that the user can quickly locate them again (see, e.g., Figure 19.4).

A more difficult form of support that still represents a challenge involves supporting the user in executing *workflows* (see, e.g., Shen, Fitzhenry, and Dietterich [2009]). That is, instead of just recognizing that the user is working on the project "quarterly report," the system (1) learns by observation what steps are involved in the preparation of a quarterly report; (2) keeps track of how much of the quarterly report workflow the user has executed so far; and (3) supports the user in remembering and executing subsequent steps. The tendency of users to multitask makes this type of support potentially valuable, but it also makes it challenging for systems to do the necessary learning and activity tracking.

Two traditionally popular candidates for automation of this general type have been sorting or filtering e-mail and scheduling appointments and meetings. Classic early research on these tasks included the work of Pattie Maes's group on "agents that reduce work and information overload" (see, e.g., Maes [1994]). The scheduling of meetings and appointments has been addressed over the years by (among many others) Mitchell et al. (1994); Horvitz (1999); and Gervasio et al. (2005): by learning the user's general preferences for particular meeting types, locations, and times of day, a system can tentatively perform part of the task of entering appointments in the user's calendar.

Systems of this sort can actually take over two types of work from the user: (1) choosing what particular action is to be performed (e.g., which folder a file should be saved in); and (2) performing the mechanical steps necessary to execute that action (e.g., clicking in the file selector box until the relevant folder has been reached). Adaptation to the user is required only for the first type of work; but the second type of work cannot be performed without the first type.

In the ideal case, the system could make the correct choice with such confidence that it would not even be necessary to consult the user, and the entire task would be automated, with the user perhaps not even being aware that it was being performed. In many cases, though, the user does have to be involved in the choice process, because the system can only help to make the choice, not make it autonomously. In these cases, the amount of mental and physical work saved is much lower. Hence, there is a trade-off between the amount of control that the user has over the choices being made and the amount of effort they save. Users can differ as to where they want to be on this trade-off curve at any given time, depending on factors like the importance of making a correct choice and the amount of other work that is competing for their attention. The typical pattern is for users to begin by exercising careful control over the performance of the task and then to relinquish control gradually to the system, as the system's competence increases (because of learning) and/or the user becomes better able to predict what the system will be able to do successfully. Patterns of this sort will be discussed in the Section 19.4.

19.2.3 Adapting the Interface to Individual Tasks and Usage

A different way of helping a person to use a system more effectively is to adapt the presentation and organization of the interface so that it fits better with the user's tasks and usage patterns. The potential benefit of this type of adaptation is that it can improve the user's motor performance by bringing functionality that is likely to be used closer or making interface elements larger; improve perceptual performance by making relevant items easier to find; or improve cognitive performance by reducing complexity.

An example of this type of adaptive interface that will be familiar to most readers is the font selection menus available in popular productivity software. Figure 19.6a illustrates the basic mechanism: the most recently selected items are copied to the top part of the menu. This top part, clearly visually separated from the rest of the menu, holds the adaptive content. If a desired font is present in the top section, the user can select it either from that section or from its usual location in the lower part of the menu.

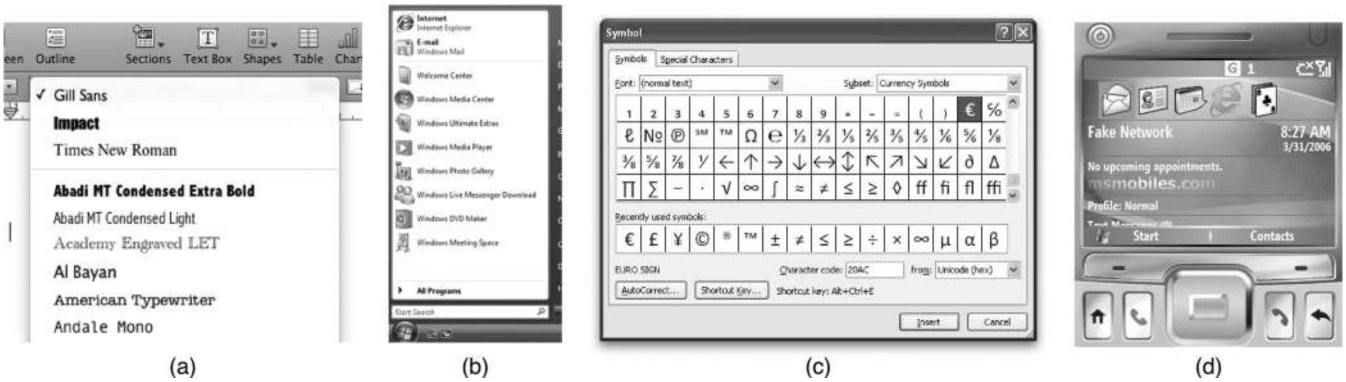


FIGURE 19.6 Examples of modern implementations of adaptive split interfaces. (a) The most recently used fonts are copied to the clearly designated adaptive top part of the menu in Apple Pages. A user wishing to select the Times New Roman font has the option of either taking advantage of the adaptation or following the familiar route to the usual location of that font in the main part of the menu. (b) Recently or frequently used programs are copied to the main part of the Windows 7 Start menu while also remaining accessible through the "All Programs" button. (c) Recently used special symbols are copied to a separate part of the dialog box in the symbol chooser in MS Office 2007. (d) Recently used applications are easily accessible on a Windows Mobile phone.

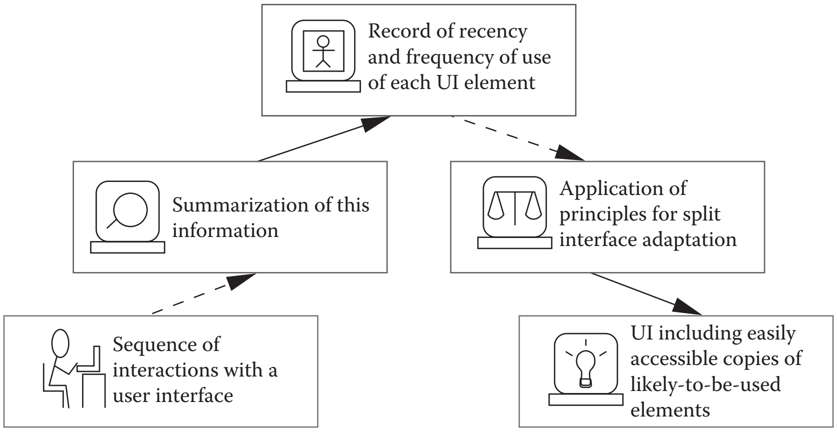

Record of recency and frequency of use of each UI element
Summarization of this information
Application of principles for split interface adaptation
Sequence of interactions with a user interface
UI including easily accessible copies of likely-to-be-used elements

FIGURE 19.7 Overview of adaptation in split interfaces.

The concept generalizes beyond menus; it can be used to adapt many different types of UI component, as is illustrated in Figure 19.6. We use the term *split interfaces* to refer to the successful general design pattern in which adaptation is used to copy functionality predicted to be most relevant to the user to a designated adaptive part of the UI (see Figure 19.7).

Several studies have demonstrated that split interfaces reliably improve both satisfaction and performance (Findlater and McGrenere 2008; Gajos et al. 2006). What makes split interfaces successful is that they offer an effort saving to those users who are willing to take advantage of the adaptation while not upsetting the familiar routine for those who prefer to use the basic interface consistently.

Designs that require users to alter their behavior are often resisted. A widely known example of an early adaptive interface that elicited mixed reactions from users is the Smart Menus that Microsoft introduced in Windows 2000 (Figure 19.8; see McGrenere, Baecker, and Booth [2007] for a comparison of this type of adaptation with user-controlled customization). To reduce the apparent complexity of the software, these menus were designed to show only a subset of the features—the most basic ones and those that the user used frequently or recently. The remaining features were shown if the user dwelled on a menu without selecting anything or if he or she clicked on a downward pointing arrow at the bottom of the menu. The design had the promise of simplifying the interaction most of the time for most users, but for some users, the additional mental effort required to find infrequently used functionality outweighed the potential benefits.

An early illustrative example involves automatically reordering menu items on the basis of the frequency of use (Mitchell and Shneiderman 1989). This approach resulted in poorer performance and lower user satisfaction than the nonadaptive baseline. In this case, the lack of success of the adaptive strategy can be attributed to the fact that because of the constantly changing order of menu items, users could never reach the expert level of visual search efficiency that is achievable with unchanging familiar interfaces.

A radically different approach to menu adaptation—called *ephemeral adaptation*—was introduced recently by Findlater et al. (2009). In ephemeral adaptation, the menu items that are predicted to be most likely to be selected by the user are displayed immediately when the menu is opened,

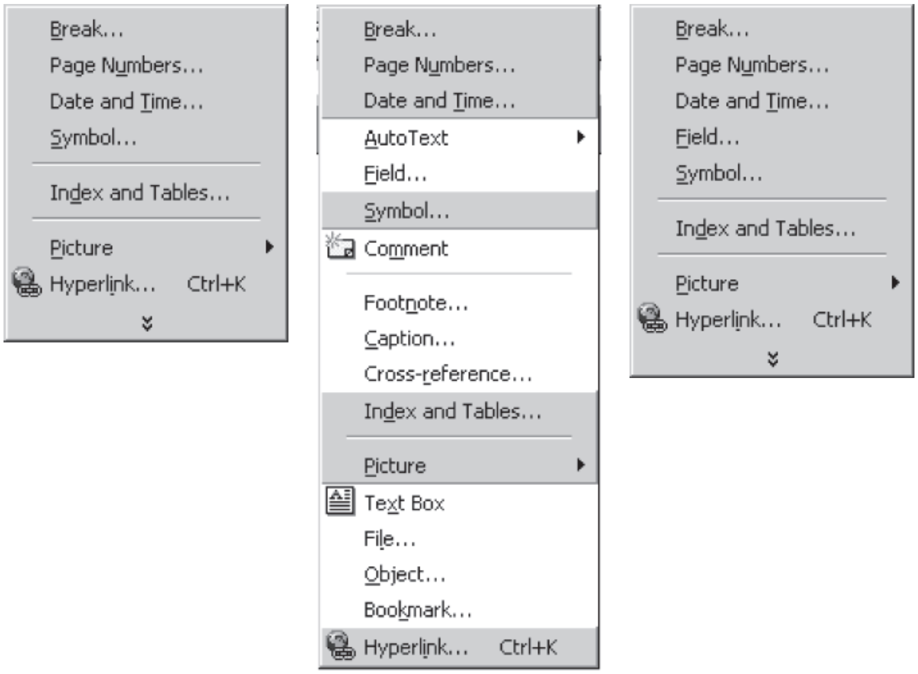


FIGURE 19.8 In Microsoft Smart Menus, rarely used items are removed from the menu thus reducing the apparent complexity of the application. (Hovering over the menu or clicking on the double arrow below the last item causes the menu to be expanded showing the elided items. If a hidden item is selected by the user, it is immediately visible on the subsequent visits to that menu.)

whereas the remaining items fade in gradually over a short period of time (e.g., 500 milliseconds). This adaptation takes advantage of the fact that an abrupt appearance of a new object involuntarily captures our attention, while its gradual onset does not. Because the user's attention is drawn to the small subset of items that are shown immediately when the menu opens, it is easy for users to locate these items quickly. This adaptive mechanism focuses entirely on users' visual search performance. It has been demonstrated to improve overall performance without increasing selection times for the items that are gradually faded in.

A user-driven alternative to the class of adaptive approaches described in this section is customization. Customization, however, requires significant upfront effort on user's part and consequently very few people choose to customize their interfaces (Mackay 1991; Palen 1999) and even fewer recustomize them as their needs change (McGrenere et al. 2007). Mixed-initiative approaches (e.g., Bunt, Conati, and McGrenere 2007) that combine the two approaches show promise for providing good balance between efficiency and user control.

The adaptive designs discussed in this section were prototyped and evaluated mostly with menus and toolbars, but the underlying concepts can be generalized to a broader range of settings. Findlater and Gajos (2009) provide a more in-depth exploration of the design space of UIs that adapt to users' tasks.

19.2.4 Adapting the Interface to Individual Abilities

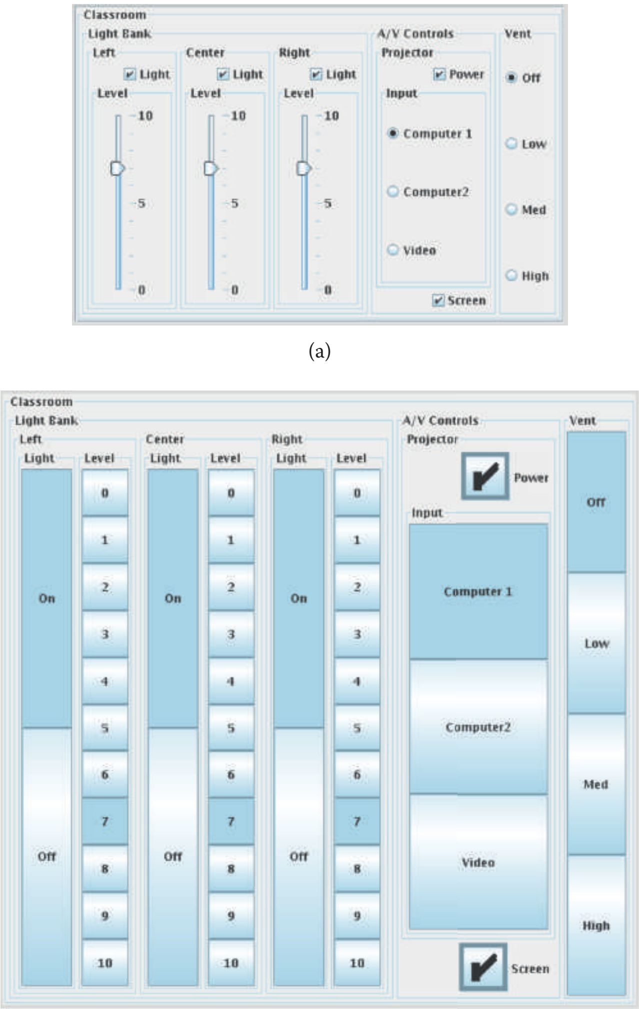
Next, we consider systems that adapt their UIs to the *abilities* of their users.

The promise of this type of adaptation is that it can provide personalized experience to people whose needs with respect to the UI are unique, variable over time, or hard to anticipate. This is precisely the situation of many users with impairments. Not only are these users different from the "average" user, they are also significantly different from each other: even people with very similar diagnoses can have very different actual abilities (Bergman and Johnson 1995; Hwang et al. 2004; Keates et al. 2002; Law, Sears, and Price 2005). Currently, these users have to adapt themselves—often using specialized assistive technologies—to the existing UIs. Adaptive systems offer the possibility to reverse this situation: Why not adapt UIs to the unique needs and abilities of people with impairments?

Impairments do not have to be permanent or a result of a medical condition. For example, environmental factors such as temperature may temporarily impair a person's dexterity; a low level of illumination will impact reading speed; and ambient noise will affect hearing ability. These factors are particularly relevant to mobile computing. Indeed, studies have shown that in relation to standing still, walking results in lower pointing speed and accuracy, as well as decreased reading speed and comprehension (Barnard et al. 2007; Lin et al. 2007). These results suggest that there is both a need and an opportunity to adapt mobile interaction to the momentary effective abilities of users.

The Supple system (Gajos, Wobbrock, and Weld 2007, 2008; Gajos, Weld, and Wobbrock 2010) provides an example of ability-based adaptation for people with motor impairments. Supple requires each user to perform a one-time set of diagnostic tasks so that the system can build a model of that person's unique motor abilities. After that, for any application that the user wants to interact with, Supple uses optimization methods to automatically generate UIs that are predicted to be the fastest to use for this person. Figure 19.9 shows an example of a dialog box automatically generated by Supple for a user with impaired dexterity due to a spinal cord injury; see also Figure 19.10. The results of an experiment involving 11 participants with a variety of motor impairments demonstrate that the automatically generated interfaces that were adapted to users' individual motor abilities resulted in significantly improved speed, accuracy, and satisfaction (see Gajos, Wobbrock, and Weld [2008]). On the average, these interfaces helped close over 60% of the performance gap between able-bodied users and users with motor impairments.

The Walking UI prototype (Kane, Wobbrock, and Smith 2008) shown in Figure 19.11 provides an example of what an adaptation to the changing abilities of mobile users might look like. The UI has two versions, one for when the user is stationary and one for when they are in motion. The two versions follow a very similar design to ensure that the users do not have to learn two separate UIs. The walking variant has larger interactors to compensate for users' impaired dexterity, larger fonts for song titles to accommodate reduced reading ability, and a more visually salient presentation for song titles than for secondary information to mitigate the effects of fragmented attention.



(a)

(b)

FIGURE 19.9 Example of the ability-based adaptation in Supple. (a) The default interface for controlling lighting and A/V equipment in a classroom. (b) A user interface for the same application automatically generated by Supple for a user with impaired dexterity based on a model of her actual motor abilities.

These types of systems have been evaluated in laboratory studies, but since they have not yet been widely deployed, we cannot yet provide empirical evidence showing what the main challenges to adoption of these systems are. But several such challenges can be anticipated: Obtaining useful models of users' abilities while placing minimum burden on the users is clearly one such challenge. The studies evaluating the Supple system demonstrated that models created from direct measurements of users' abilities resulted in significantly more successful interfaces than those that were based on users' expressed preferences, but those direct measurements of abilities required users to go through a one-time but hour-long set of diagnostic tasks. Another factor that seems likely to have an impact on adoption of interfaces like that of Figure 19.11 is the method for controlling the switch between different UI variants. A fully manual approach is likely to be found too inefficient, whereas one that is fully automated may cause confusion.

Wobbrock et al. (2011) present several other examples of ability-based interfaces, discuss the rationale for ability-based design, and propose a set of guiding principles.

19.3 FUNCTIONS: SUPPORTING INFORMATION ACQUISITION

Even back in the days when computers were chained to desktops, people were complaining about information overload and clamoring for tools that would help them to focus their attention on the documents, products, and people that really mattered to them. Since then, the flood has grown to a tsunami. Two of the most conspicuous developments have been (1) mobile devices that enable people to produce and consume information wherever they are; and (often in combination with these) (2) social networks, which are increasingly replacing face-to-face communication.

This information overload constitutes a powerful motivation for the development of systems that adapt to their users: computers have the technical capability to reduce the information tsunami to a trickle that people can manage; but since people are generally not interested in the same trickle, computers can do so effectively only by taking into account properties of the user such as their interests, current tasks, and context.

19.3.1 Helping Users to Find Information

We will first look at the broad class of systems that help the user to find relevant electronic documents, which may range from brief news stories to complex multimedia objects.

One type of document that has become more pervasive over the past several years comprises news reports of the type traditionally found in printed newspapers. With so many

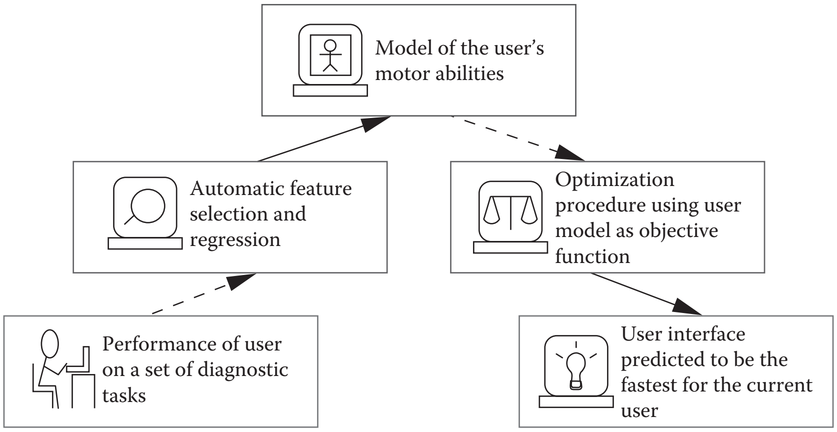
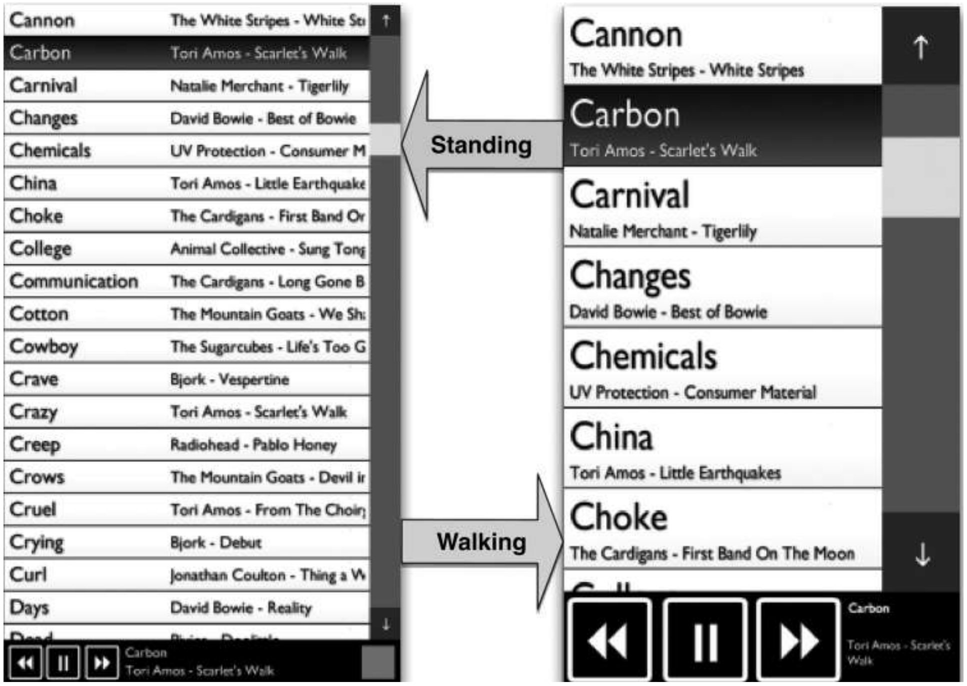

Model of the user's motor abilities
Automatic feature selection and regression
Optimization procedure using user model as objective function
Performance of user on a set of diagnostic tasks
User interface predicted to be the fastest for the current user

FIGURE 19.10 Overview of adaptation in SUPPLE.



Cannon	The White Stripes - White St
Carbon	Tori Amos - Scarlet's Walk
Carnival	Natalie Merchant - Tigerlily
Changes	David Bowie - Best of Bowie
Chemicals	UV Protection - Consumer M
China	Tori Amos - Little Earthquake
Choke	The Cardigans - First Band Or
College	Animal Collective - Sung Tong
Communication	The Cardigans - Long Gone B
Cotton	The Mountain Goats - We Sh
Cowboy	The Sugarcubes - Life's Too G
Crave	Bjork - Vespertine
Crazy	Tori Amos - Scarlet's Walk
Creep	Radiohead - Pablo Honey
Crows	The Mountain Goats - Devil ir
Cruel	Tori Amos - From The Choir
Crying	Bjork - Debut
Curl	Jonathan Coulton - Thing a W
Days	David Bowie - Reality

Carbon
Tori Amos - Scarlet's Walk

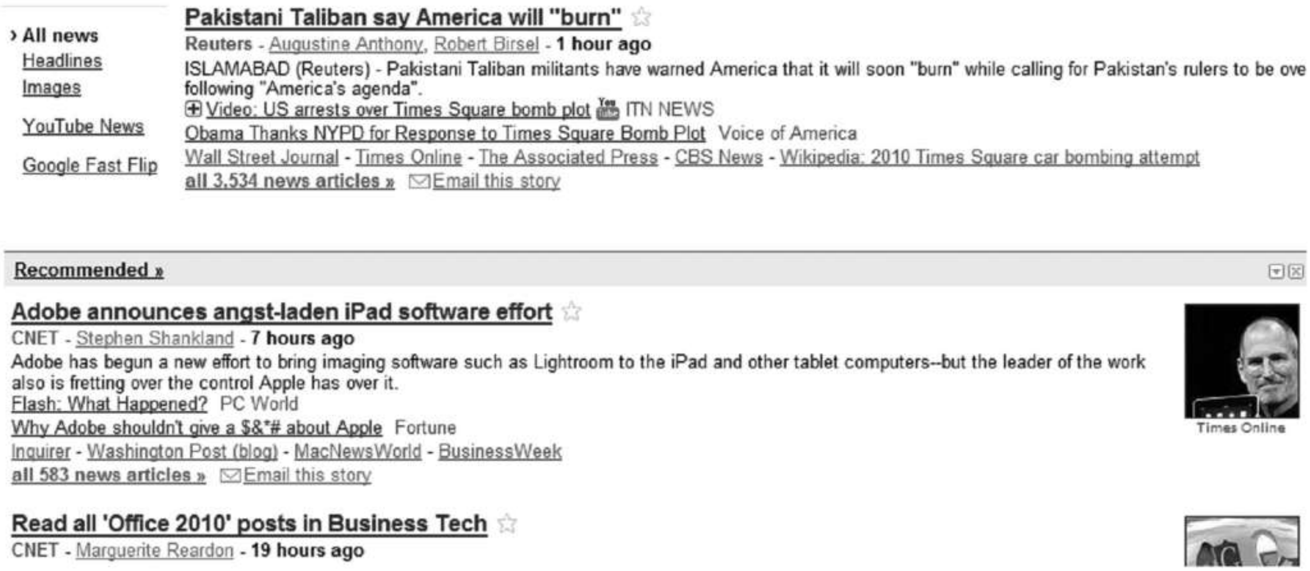
Cannon	The White Stripes - White Stripes
Carbon	Tori Amos - Scarlet's Walk
Carnival	Natalie Merchant - Tigerlily
Changes	David Bowie - Best of Bowie
Chemicals	UV Protection - Consumer Material
China	Tori Amos - Little Earthquakes
Choke	The Cardigans - First Band On The Moon

Carbon
Tori Amos - Scarlet's Walk

FIGURE 19.11 The Walking User Interface—an example of an adaptation to a temporary situationally induced impairment. The larger buttons address the decreased pointing speed and accuracy of walking users; the larger fonts for song titles help with impaired reading speed; and the differences in font sizes between titles and additional song information help direct fragmented attention. (Screenshots courtesy of Shaun Kane.)

news sources now available online, the amount of choice available to a person who wants to read a few interesting news reports has increased drastically, even when we take into consideration the fact that one report may turn up in a number of variations in different news sources.

One news website that addresses this problem is GOOGLE NEWS. One of the solutions offered by the site is a section of "recommended" stories that are selected on the basis of the users' previous clicks on other news stories within the same site (see Figures 19.12 and 19.13). The first personalization algorithms used for this purpose by GOOGLE were based on collaborative filtering, which is found in several other systems discussed in this chapter. But as is described by Liu, Dolan, and Pedersen (2010), it proved necessary to include some *content-based* filtering as well, recommending stories related to general themes that the current user had previously shown an interest in. In particular, it is otherwise hard to recommend hot-off-the-press news stories that have not yet attracted many clicks from other users. Roughly speaking, in this application, the content-based filtering helps by revealing what topics the current user is generally interested in, whereas the collaborative filtering helps to keep track of temporary trends (e.g., a surge of interest in the newly released iPad) that apply to larger groups of users and that are likely to be followed to some extent by any given individual user as well.



› All news
Headlines
Images
YouTube News
Google Fast Flip

Pakistani Taliban say America will "burn" ☆
Reuters - Augustine Anthony, Robert Birsel - **1 hour ago**
ISLAMABAD (Reuters) - Pakistani Taliban militants have warned America that it will soon "burn" while calling for Pakistan's rulers to be ove
following "America's agenda".
⊞ Video: US arrests over Times Square bomb plot ITN NEWS
Obama Thanks NYPD for Response to Times Square Bomb Plot Voice of America
Wall Street Journal - Times Online - The Associated Press - CBS News - Wikipedia: 2010 Times Square car bombing attempt
all 3,534 news articles » ✉Email this story

Recommended »

Adobe announces angst-laden iPad software effort ☆
CNET - Stephen Shankland - **7 hours ago**
Adobe has begun a new effort to bring imaging software such as Lightroom to the iPad and other tablet computers--but the leader of the work also is fretting over the control Apple has over it.
Flash: What Happened? PC World
Why Adobe shouldn't give a $&*# about Apple Fortune
Inquirer - Washington Post (blog) - MacNewsWorld - BusinessWeek
all 583 news articles » ✉Email this story

Times Online

Read all 'Office 2010' posts in Business Tech ☆
CNET - Marguerite Reardon - **19 hours ago**

FIGURE 19.12 A small section of a front page of Google News, including the personalized section. (This user had recently selected a number of computer-related articles, and at this time—May 2010—the recently launched iPad was a popular topic.)

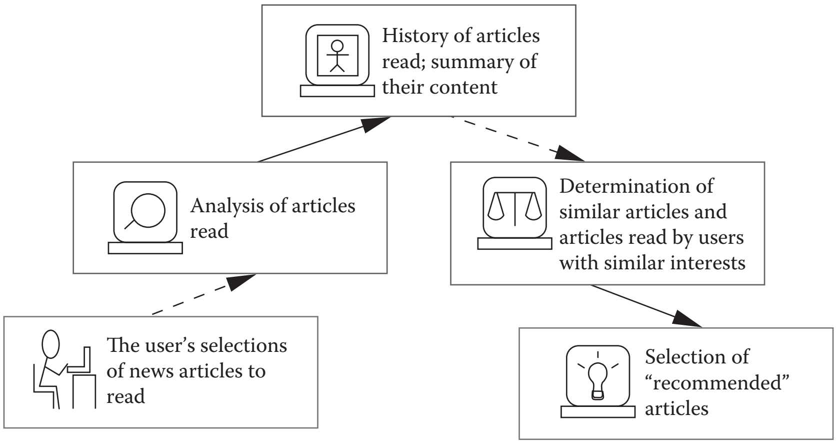

History of articles read; summary of their content
Analysis of articles read
Determination of similar articles and articles read by users with similar interests
The user's selections of news articles to read
Selection of "recommended" articles

FIGURE 19.13 Overview of adaptation in Google's personalized news recommendations.

More generally speaking, user-adaptive systems that help users find information* typically draw from the vast repertoire of techniques for analyzing textual information (and to a lesser but increasing extent, information presented in other media) that have been developed in the field of information retrieval. The forms of adaptive support are in part different in three different situations, the first two of which can arise with Google News.

19.3.1.1 Support for Browsing

In the world wide web and other hypermedia systems, users often actively search for desired information by examining information items and pursuing cross-references among them. A user-adaptive hypermedia system can help focus the user's browsing activity by recommending or selecting promising items or directions of search on the basis of what the system has been able to infer about the user's information needs. An especially attractive application scenario is that of mobile information access, where browsing through irrelevant pages can be especially time consuming and expensive. In this context, the best approach may be for the system to omit entirely links that it expects to be less interesting to the individual user. Billsus and Pazzani (2007) describe a case study of an adaptive news server that operated in this way. Stationary systems with greater communication bandwidth tend to include all of the same links that would be presented by a nonadaptive system, highlighting the ones that they consider most likely to be of interest or presenting separate lists of recommended links. As is argued and illustrated by Tsandilas and schraefel (2004), this approach makes it easier for the user to remedy incorrect assessments of the user's interests on the part of the system.

* Surveys of parts of this large area are provided by, among others, Kelly and Teevan (2003) and several chapters in the collection edited by Brusilovsky, Kobsa, and Nejdl (2007).

19.3.1.2 Support for Query-Based Search

When a user is just checking the latest news or casually browsing for interesting information, the user is not in general expressing a specific information need. Hence, it is relatively easy for a user model to help noticeably by presenting information that is especially likely to be of interest to this particular user. In contrast, when a user formulates an explicit query, as in a web search engine, it is less obvious how a user model can help to identify relevant information. And in fact, the *potential for personalization* (Teevan, Dumais, and Horvitz 2010) has been found to vary considerably from one query to the next. If just about all of the users who issue a given query end up choosing the same documents from those returned by the system, there is little that an individual user model can do to increase the usefulness of the search results. But for queries that tend to result in very different selections for different users (e.g., the query "chi"), *personalized search* can add value. The basic idea is that the list of search results that would normally be returned is reordered (or *biased*) on the basis of a user model, which is in turn based on some aspects of the user's previous behavior with the system. GOOGLE has offered personalized search on its main search engine for several years—though many users are probably unaware of the personalization, which tends not to change the ranking of the search results in an immediately noticeable way for most queries.

The idea of assessing the potential for personalization is worth considering with other forms of adaptation as well: if we can estimate in advance the possible benefits of adaptation, perhaps before designing or implementing any adaptive mechanism, we can more efficiently identify situations in which the benefits of adaptation will outweigh the costs.

19.3.1.3 Spontaneous Provision of Information

A number of systems present information that may be useful to the user even while the user is simply working on some task, making no effort to retrieve information relevant to it from external sources. An illustrative recent example* is the AMBIENT HELP system (Matejka, Grossman, and Fitzmaurice 2011), which can also be seen as an approach to the problem of offering adaptive help that was discussed at the beginning of this chapter: while a user works with a complex application on one computer monitor, AMBIENT HELP uses a second monitor to display videos and texts with tutorial material that has some relevance to the user's current working context. A central design issue for this and similar systems concerns the methods for making the retrieved information available to the user. Presentation of results via means like pop-up windows risks being obtrusive (cf. Section 19.4); but if the presentation is too subtle, users will often ignore the information that is offered and derive little or no benefit from the system. AMBIENT HELP expands the space of design solutions by introducing an unobtrusive way of showing what a video has to offer (with a dimmed image, a reduced frame rate, and muted volume) and a scheme for allowing users quickly to explore the content of the available videos. Previous work in the same vein (e.g., by Billsus, Hilbert, and Maynes-Aminzade [2005]) suggests allowing users to adjust the relative obtrusiveness of the proactively offered information to suit their individual taste.

19.3.2 RECOMMENDING PRODUCTS

One of the most practically important categories of user-adaptive systems today comprises the product recommenders that are found in many commercial websites. The primary benefit of these systems is that they assist users in finding personally interesting (but not previously known) items in large collections of products.

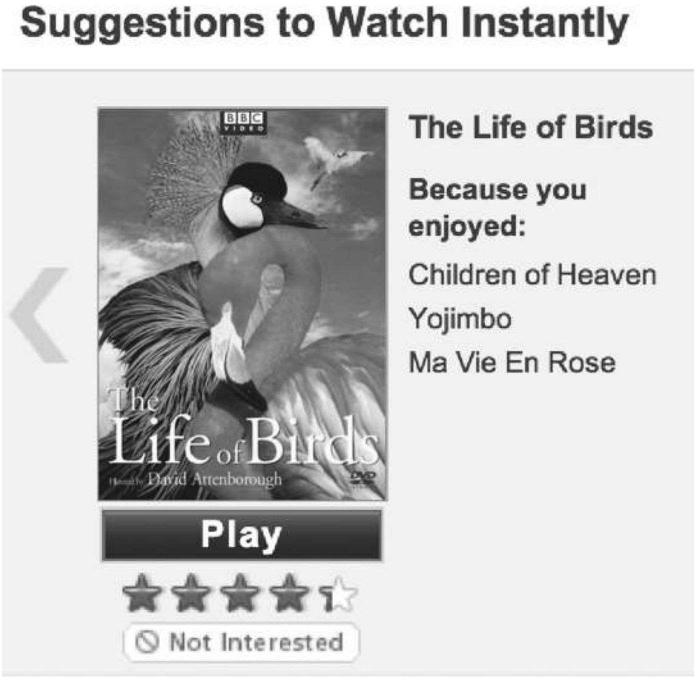
An example that will look familiar to most readers is shown in Figure 19.14. A visitor to NETFLIX has just explicitly requested recommendations, without having specified a particular type of a movie. During the user's past visits, NETFLIX has learned about his or her interests, on the basis of movies he or she has watched and ratings he or she has made. Therefore, the system can make recommendations that are especially likely to appeal to this particular user.

The recommendations of NETFLIX embody many design decisions that contribute to the success of this type of adaptation. First, as can be inferred from the brief explanation that accompanies the recommendation in Figure 19.14, the system takes as a starting point the information it has about the user's prior viewing history and ratings. It then compares these with the ratings of other users to generate predictions for the current user. That is, the recommendations are based on a statistical analysis of ratings made by many users, an approach known as *collaborative filtering* (see, e.g., Schafer et al. [2007], for a general overview and Figure 19.15).[†] The products recommended in this way may also happen to be similar in the sense of belonging to the same genre or having the same director, but similarities of this sort can also be conspicuously absent: In the example shown in Figure 19.14, a nature documentary is recommended to a customer partly on the basis of his past enjoyment of Kurosawa's lighthearted samurai story *Yojimbo*. The power of collaborative filtering comes from the fact that many features relevant to our choices are hard to capture. In the movie domain, for example, the mood, the particular style of humor, or the details of the camera work may be as relevant as the more easily describable properties such as genre, director, or cast.

Second, the explanations accompanying the recommendation are another important design feature: for example, taking into account the fact that what is "good" often depends on context (e.g., a user may enjoy a complex drama one day while preferring a less demanding action movie after a long work day), the explanations help users better predict if a particular film is what they are looking for at a given moment. As in the example in Figure 19.14, many sites use other items

[*] Influential early systems in this category include those of Rhodes (2000) and Budzik, Hammond, and Birnbaum (2001).

[†] Interested readers will also find many documents available on the web about the highly publicized efforts of NETFLIX to encourage improvement of its algorithms by sponsoring the NETFLIX Prize.



Suggestions to Watch Instantly

The Life of Birds

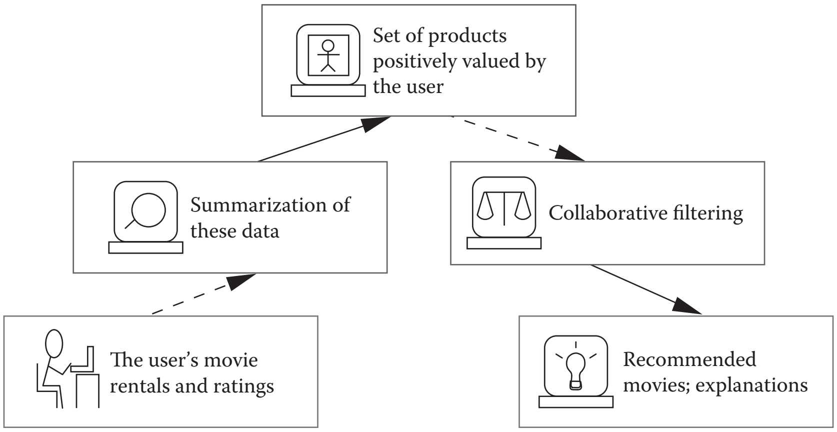
Because you enjoyed:

Children of Heaven

Yojimbo

Ma Vie En Rose

FIGURE 19.14 Part of a screen showing a movie recommendation generated on request by NETFLIX. (Screen shot made from http://netflix.com/ in March 2010.)



Set of products positively valued by the user
Summarization of these data
Collaborative filtering
The user's movie rentals and ratings
Recommended movies; explanations

FIGURE 19.15 Overview of adaptation in NETFLIX.

the user has rated in the past as a basis for generating an explanation. But very different types of information can also be used for explanations, such as user-generated tags (Vig, Sen, and Riedl 2009). A recent discussion of the many forms that explanations can take and the functions that they can serve has been provided by Tintarev and Masthoff (2010).

Finally, movie recommendations in NETFLIX complement rather than replace the normal searching and browsing capabilities. This property allows users to decide which mode of interaction is most appropriate in their situation.

Because many products, such as movies, vacations, or restaurant meals, are often enjoyed by groups rather than individuals, so a number of systems have been developed that explicitly address groups (see Jameson and Smyth [2007] for an overview). The need to address a group rather than an individual has an impact on several aspects of the recommendation process: Users may want to specify their preferences in a collaborative way; there must be some appropriate and fair way of combining the information about the various users' preferences; the explanations of the recommendations may have to refer to the preferences of the individual group members; and it may be worthwhile for the system to help the users negotiate to arrive at a final decision on the basis of the recommendations.

Another design challenge for recommender systems has to do with the availability of information about the users' preferences. Collaborative filtering is less effective for supporting infrequent decisions such as a digital camera purchase, which can involve one-time considerations that are not closely related to previous choices by the same user. Since the 1980s, researchers have worked on systems that explicitly elicit information about the user's needs (and the trade-offs among them) and help the user identify products that best meet their needs. One particularly effective interaction paradigm for such systems is *example critiquing* (see, e.g., Burke, Hammond, and Young [1997], for an early exposition, and Pu and Chen [2008] for a discussion of some recent advances). The distinguishing feature is an iterative cycle in which the system proposes a product (e.g., a restaurant in a given city), the user criticizes the proposal (e.g., asking for a "more casual" restaurant), and the system proceeds to propose a similar product that takes the critique into account.

Finally, a highly pervasive and economically vital form of product recommendation is advertising. Over the last decade, online advertising has shifted largely from attention-grabbing banners and pop-ups to subtler *personalized* ads. Rather than relying on users' explicit feedback in the form of purchases and product ratings (as is the case with recommender systems), online personalized advertising relies on implicit input such as the search terms, contents of an e-mail message (in the case of GMAIL ads), the topics of the pages visited, and the browsing history. There are many good reasons to prefer such personalized advertising: it tends to be presented in a less intrusive way (e.g., the text-only ads used by GOOGLE), and it has the promise of being more relevant to the users. Indeed, a recent study found that of people who clicked on personalized ads, twice as many were likely to actually make a purchase than people who clicked on nonpersonalized ads (Beales 2010).

However, because online behavioral data (such as searches the people perform and sites that they visit) are considered sensitive personal information, and because the users do not have clear and effective means of controlling what information they divulge to advertisers and when, privacy concerns about personalized advertising are common (Federal Trade Commission 2009). The Canadian Marketing Association (2009) has found that only about 30% of North Americans are comfortable with advertisers tracking their browsing behavior for the purpose of providing more targeted advertising, even though nearly half like seeing ads for coupons and promotions from online stores and brands that they have purchased from before. Improving the comprehensibility of and user control over data collection are therefore important challenges for the long-term success of personalized advertising (cf. in Section 19.4).

19.3.3 Tailoring Information Presentation

Sections 19.3.1–2 discussed systems that help users decide *what* information (such as news items or product descriptions) to consider. We now turn to systems that adapt *how* information is presented.

A striking and practically important example is found in the work of Jefferson and Harvey (2006, 2007), which uses personalized models of color perception abilities of color-blind users to adapt the presentation of graphical information in a way that preserves the saliency and readability of color-encoded information. A major challenge in adapting content to an individual's color perception abilities is that complex color-encoded information needs to be conveyed through a reduced color palette. One possible approach is to generate a fixed mapping that tries to "squeeze" the full spectrum of visible colors into a range that is distinguishable by a particular individual. This approach inevitably reduces perceptual differences among the colors in the transformed palette. Instead, Jefferson and Harvey (2006) compute these mappings for each image individually (see Figure 19.16). Their approach takes advantage of the fact that most images use only a limited number of colors for salient information. Their algorithm automatically identifies these salient colors and computes a mapping from the original palette to one that is appropriate for the user. The mapping is computed in a way that preserves the perceptual differences among the important colors.

Unfortunately, because the process of computing an optimal color mapping is computationally expensive—up to several minutes may be required—it is not feasible for interactive use. Jefferson and Harvey (2007) have developed an alternative approach where the computer quickly generates a small set of possible mappings that may be appropriate for a particular individual and the user can quickly select the appropriate one with a slider, while getting an immediate preview of the effect. By splitting the adaptation burden

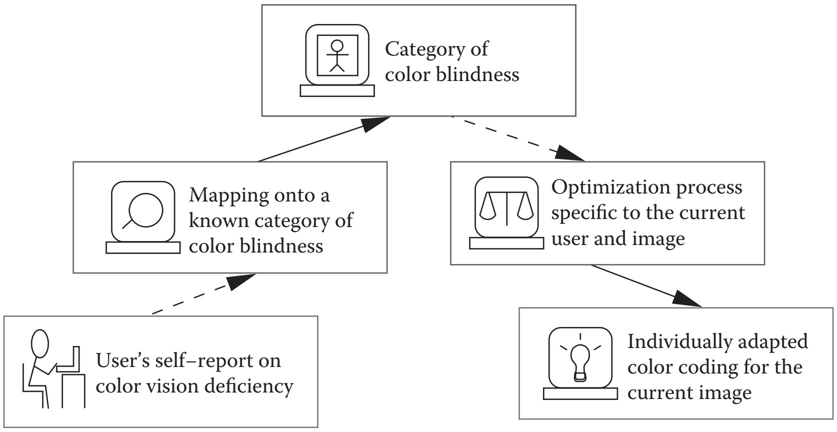

Category of color blindness
Mapping onto a known category of color blindness
Optimization process specific to the current user and image
User's self–report on color vision deficiency
Individually adapted color coding for the current image

FIGURE 19.16 Overview of Jefferson and Harvey's method of adaptation to color blindness.

between the computer and the user, this particular system provides users with a solution that is effective and fast, and requires only a minimal amount of user effort.

The remaining challenge is that of quickly creating accurate models of individual color perception abilities. Fortunately, most users can be helped adequately by being stereotyped into one of a small number of discrete color blindness categories. However, some types of color blindness (the *anomalous trichromacies*) form a spectrum from almost normal color perception to almost complete inability to distinguish certain pairs of colors. Although there exist some methods for building models of individual color perception abilities (e.g., Brettel, Viénot, and Mollon [1997]; Gutkauf, Thies, and Domik [1997]), they require that users engage in an explicit diagnostic task, and one that may need to be repeated for different display devices. A faster, unobtrusive method is still needed.

Tailoring often concerns information in textual form. An important application area here comprises systems that present medical information to patients, who may differ greatly in terms of their interest in and their ability to understand particular types of information (see, e.g., Cawsey, Grasso, and Paris [2007], for an overview).

Properties of users that may be taken into account in the tailoring of documents include: the user's degree of interest in particular topics; the user's knowledge about particular concepts or topics; the user's preference or need for particular forms of information presentation; and the display capabilities of the user's computing device (e.g., web browser vs. cell phone).

Even in cases where it is straightforward to determine the relevant properties of the user, the automatic creation of adapted presentations can require sophisticated techniques of natural language generation (see, e.g., Bontcheva and Wilks [2005]) and/or multimedia presentation generation. Various less complex ways of adapting hypermedia documents to individual users have also been developed (see Bunt, Carenini, and Conati [2007] for a broad overview).

19.3.4 Bringing People Together

One of the most striking changes in computer use over the past several years has been the growth of social networks. Whereas people used to complain about being overwhelmed by the number of e-mails and other documents that they were expected to read, they can now also be overwhelmed by the number of comments posted on their social network home page, the number of people who would like to link up with them—and even the suggestions that they get from sites like Facebook and LinkedIn concerning possible social links. Accordingly, personalized support for decisions about whom to link up with has become a practically significant application area for user-adaptive systems.

Figure 19.17 shows how an internal social networking site used at IBM called SocialBlue (formerly Beehive) recommends a colleague who might be added to the user's network (see also Figure 19.18).

As the example illustrates, SocialBlue makes extensive use of information about social relationships to arrive at recommendations: not just information about who is already explicitly linked with whom in the system (which is used, for example, on Facebook) but also types of implicit information that are commonly available within organizations, such as organizational charts and patent databases.

As described by Chen et al. (2009), SocialBlue also uses information about the similarity between two employees (e.g., the overlap in the words used in available textual descriptions of them).

These authors found that these two types of information tend to lead to different recommendations, which in turn are accepted or rejected to differing extents and for different reasons. For example, information about social relationships works better for finding colleagues that the current user already knows (but has not yet established a link to in the system), while information about similarity is better for finding promising unknown contacts.

Taking the analysis of the same data a step further, Daly, Geyer, and Millen (2010) showed that different algorithms can also have different consequences for the structure of the social network in which they are being used. For example, a system that recommends only "friends of friends" will tend to make the currently well-connected members even better connected. This result illustrates why it is often worthwhile to consider not only how well an adaptive algorithm supports a user in a typical individual case but also what its broader, longer-term consequences may be.

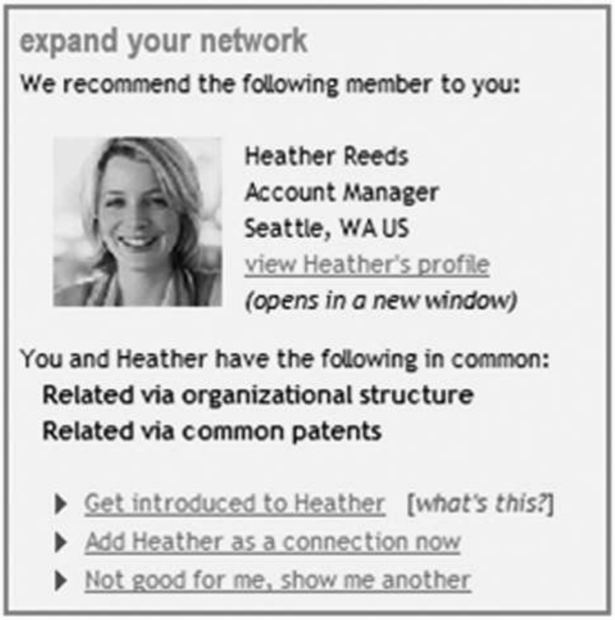
Given that the various contact recommendation algorithms can be used in combination in various ways, a natural conclusion is that designers of systems of this sort should consider what mix of the algorithm types makes most sense for their particular system and application scenario.

Other contexts in which some sort of social matching has proved useful include the following:

- Expert finding, which involves identifying a person who has the knowledge, time, and social and spatial proximity that is necessary for helping the user to solve a particular problem (see, e.g., Shami et al. [2007]; Ehrlich, Lin, and Griffiths-Fisher [2007]; Terveen and McDonald [2005]).
- Recommendation of user communities that a user might like to join—or at least use as an information resource (see, e.g., Chen, Zhang, and Chang [2008], Carmagnola, Vernero, and Grillo [2009], and Vasuki et al. [2010] for early contributions to this relatively novel problem).
- Collaborative learning, which has become a popular approach in computer-supported learning environments (see, e.g., Soller [2007]).

19.3.5 Supporting Learning

Some of the most sophisticated forms of adaptation to users have been found in tutoring systems and learning environments: systems designed to supplement human teaching by



expand your network

We recommend the following member to you:

Heather Reeds
Account Manager
Seattle, WA US
view Heather's profile
(opens in a new window)

You and Heather have the following in common:

Related via organizational structure
Related via common patents

▶ Get introduced to Heather [*what's this?*]
▶ Add Heather as a connection now
▶ Not good for me, show me another

FIGURE 19.17 Screenshot of SocialBlue showing how it recommends a potentially interesting colleague. (Image retrieved from http://www-users.cs.umn.edu/~jilin/projects.html in March 2011; reproduced with permission of Jilin Chen and Werner Geyer.)

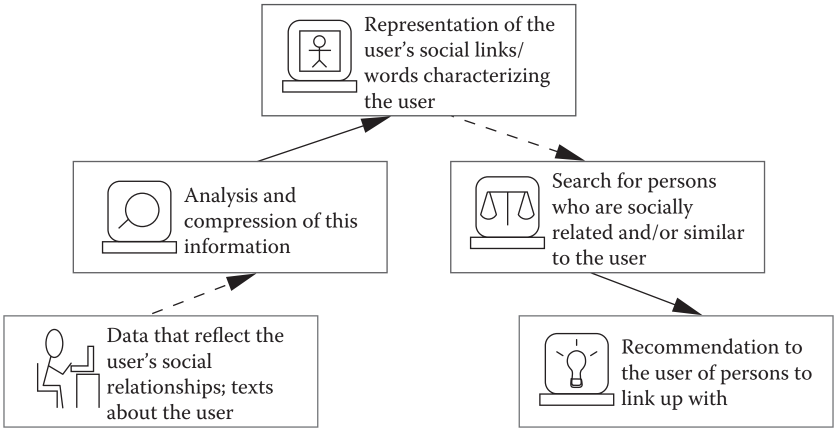
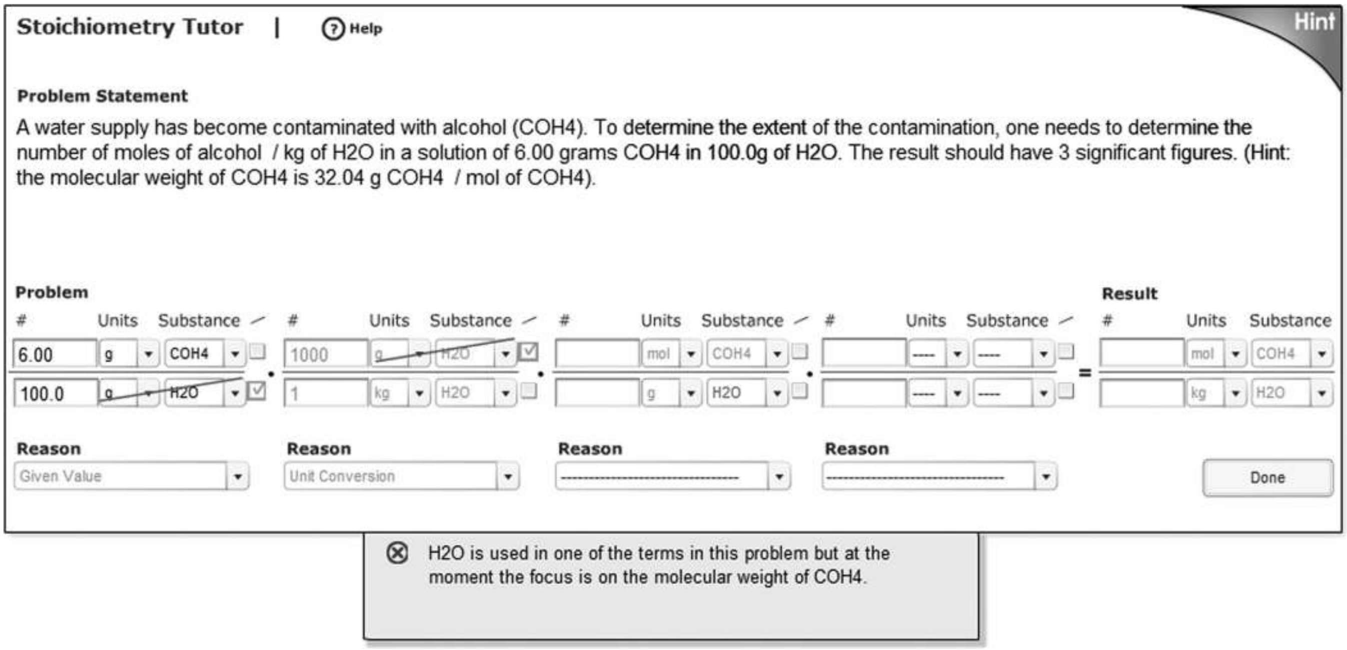

Representation of the user's social links/ words characterizing the user
Analysis and compression of this information
Search for persons who are socially related and/or similar to the user
Data that reflect the user's social relationships; texts about the user
Recommendation to the user of persons to link up with

FIGURE 19.18 Overview of adaptation in SocialBlue.

enabling students to learn and practice without such teaching while still enjoying some of its benefits.*

An illustrative recent example is the web-based Stoichiometry Tutor (Figures 19.19 and 19.20; McLaren, DeLeeuw, and Mayer 2011a,b), which helps students to practice solving elementary chemistry problems using basic mathematics. In the example shown, the student must perform a unit conversion and take into account the molecular weight of alcohol. The interface helps to structure the student's thinking, but it is still possible to make a mistake, as the student in the example has done by selecting "H20" instead of "COH4" in the lower part of the middle column. Part of the system's adaptation consists in hints that it gives when the student makes a mistake (or clicks on the "Hint" link in the upper right). The key knowledge that underlies the adaptation is a *behavior graph* for each problem: a representation of acceptable paths to a solution of the problem, along with possible incorrect steps. Essentially, the tutor is like a navigation system that knows one or more ways of getting from a specified starting point to a destination; but instead of showing the student a "route" to follow, it lets the user try to find one, offering hints when the student makes a wrong turn or asks for advice. This approach enables the system to adapt

* General sources of literature on this type of system include the *International Journal of Artificial Intelligence in Education* and the proceedings of the alternating biennial conferences on *Artificial Intelligence in Education* and on *Intelligent Tutoring Systems*. The integrative overview by VanLehn (2006) can also serve as an introduction.



Stoichiometry Tutor | ? Help

Hint

Problem Statement

A water supply has become contaminated with alcohol (COH4). To determine the extent of the contamination, one needs to determine the number of moles of alcohol / kg of H2O in a solution of 6.00 grams COH4 in 100.0g of H2O. The result should have 3 significant figures. (Hint: the molecular weight of COH4 is 32.04 g COH4 / mol of COH4).

Problem

#	Units	Substance	⁄		#	Units	Substance	⁄		#	Units	Substance	⁄		#	Units	Substance	⁄		#	Units	Substance
6.00	g	COH4	☐	•	1000	~~g~~	~~H2O~~	☑	•		mol	COH4	☐	•		----	----	☐	=		mol	COH4
100.0	~~g~~	~~H2O~~	☑		1	kg	H2O	☐			g	H2O	☐			----	----	☐			kg	H2O

Result (last three columns)

Reason	Reason	Reason	Reason	
Given Value	Unit Conversion	------------------------------	------------------------------	Done

☒ H2O is used in one of the terms in this problem but at the moment the focus is on the molecular weight of COH4.

FIGURE 19.19 Example of error feecback provided by the Stoichiometry Tutor. (The message below the main panel is the feedback on the student's incorrect selection of "H2O" as the "Substance" in the middle column, shown in red in the interface. Captured in February, 2011, from the tutor on http://learnlab.web.cmu.edu/~pact/chemstudy/learn/tutor2.html; reproduced with permission of Bruce McLaren.)

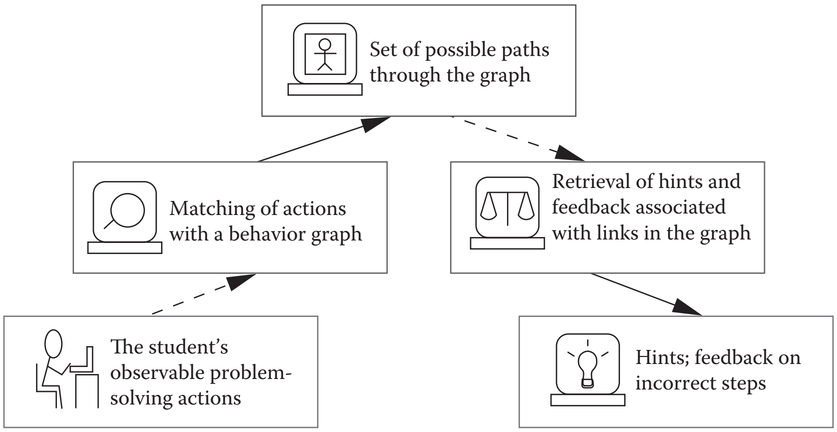

Set of possible paths through the graph
Matching of actions with a behavior graph
Retrieval of hints and feedback associated with links in the graph
The student's observable problem-solving actions
Hints; feedback on incorrect steps

FIGURE 19.20 Overview of adaptation in the Stoichiometry Tutor.

with some flexibility: It can deal with multiple strategies for solving the problem and entertain multiple interpretations about the student's behavior.

This relatively recent approach to tutoring is called *example tracing* (Aleven et al. 2009), because it involves tracing the student's progress through the behavior graph, which in turn represents, in generalized form, a set of examples of how a problem can be solved. For authors of tutoring systems, providing such examples is a relatively easy, practical way to give the system the knowledge that it needs to interpret the student's behavior. In the long history of systems that adaptively support learning, most systems have used more complex representations of the to-be-acquired knowledge and of the student's current state of knowledge (e.g., in terms of sets of rules or constraints). Example tracing is an instance of a general trend to look for simpler but effective ways of achieving useful adaptation, relative to the often complex ground-breaking systems that are developed in research laboratories.

Giving feedback and hints about steps in solving a problem is an example of *within-problem guidance*, sometimes called the *inner loop* of a tutoring system (VanLehn 2006). Adaptation can also occur in the *outer loop*, where the system makes or recommends decisions about what problems the student should work on next. Outer-loop adaptation can use coarse- or fine-grained models of the student's knowledge, which are typically constructed on the basis of observation of the student's behavior.

19.4 USABILITY CHALLENGES

One of the reasons why the systems discussed in the first part of this chapter have been successful is that they have managed to avoid some typical usability side effects that can be caused by adaptation. These side effects were quite pronounced in some of the early user-adaptive systems that came out of research laboratories in the 1980s and 1990s, and they led to some heated discussion about the general desirability of adaptation to users (see the references given later in this section). By now, designers of user-adaptive systems have learned a good deal about how to avoid these side effects, but it is still worthwhile to bear them in mind, especially when we design new forms of adaptation that go beyond mere imitation of successful existing examples.

Figure 19.21 gives a high-level summary of many of the relevant ideas that have emerged in discussions of usability issues raised by user-adaptive systems and interactive intelligent systems more generally (see, e.g., Norman [1994]; Wexelblat and Maes [1997]; Höök [2000]; Tsandilas and Schraefel [2004]; Jameson [2009]). The figure uses the metaphor of signs that give warnings and advice to persons who enter a potentially dangerous terrain.

The *usability threats* shown in the third column characterize the five most important potential side effects. A first step toward avoiding them is to understand why they can arise; the column *typical properties* lists some frequently encountered (though not always necessary) properties of user-adaptive systems, each of which has the potential of creating particular usability threats.

Each of the remaining two columns shows a different strategy for avoiding or mitigating one or more usability threats: Each of the *preventive measures* aims to ensure that one of the *typical properties* is not present in such a way that it would cause problems. Each of the *remedial measures* aims to ward off one or more threats once it has arisen. The classes of preventive and remedial measures are open ended, and in fact advances in design and research often take the form of new measures in these classes. Therefore, Figure 19.21 can be used not only as a summary of some general lessons but also as a way of structuring thinking about a specific user-adaptive system; in the latter case, some of the boxes and arrows will be replaced with content that is specific to the system under consideration.

A discussion of all of the relationships indicated in Figure 19.21 would exceed the scope of this chapter, but some remarks will help to clarify the main ideas.

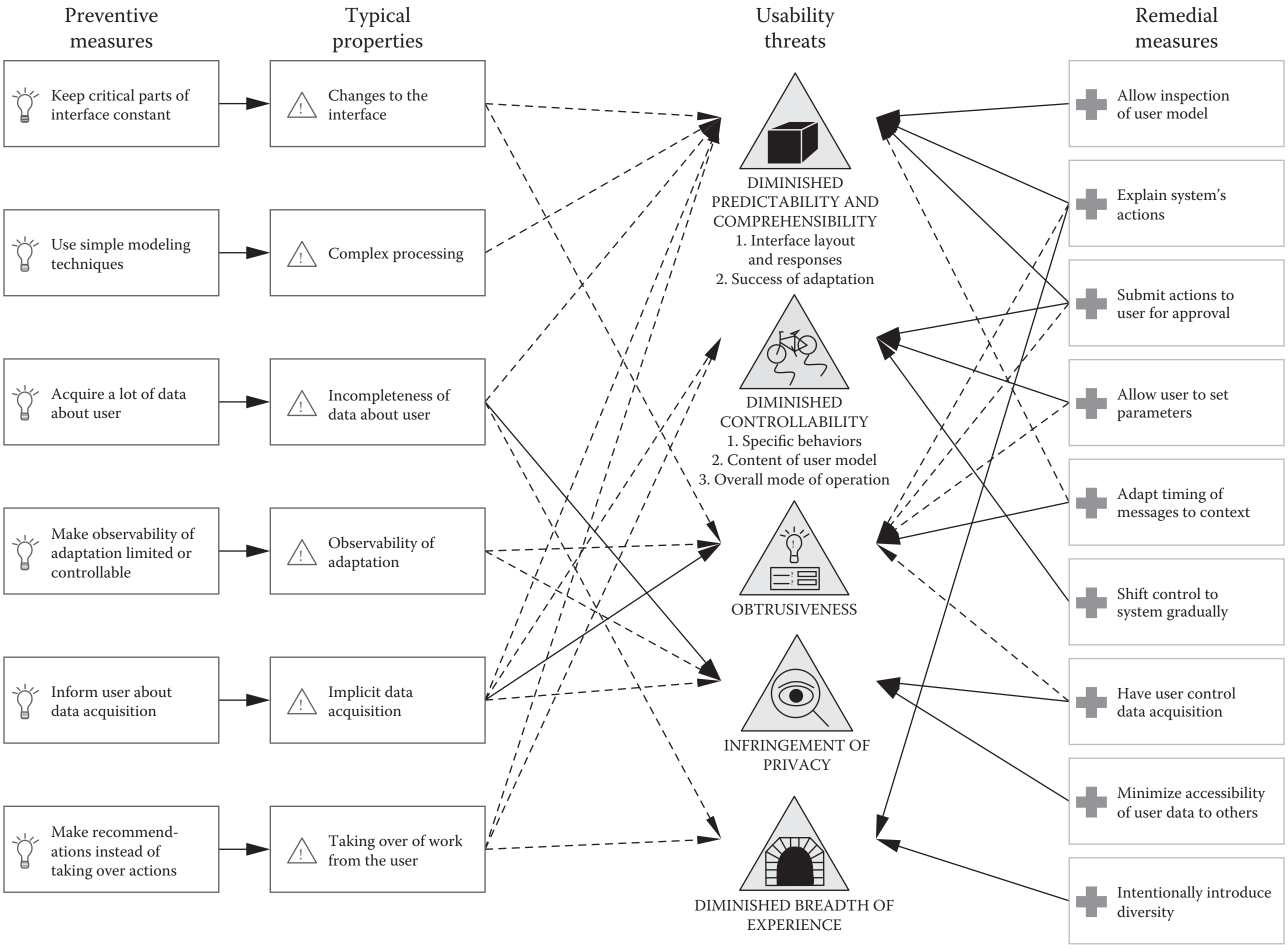

Preventive measures
Typical properties
Usability threats
Remedial measures
Keep critical parts of interface constant
Changes to the interface
Use simple modeling techniques
Complex processing
Acquire a lot of data about user
Incompleteness of data about user
Make observability of adaptation limited or controllable
Observability of adaptation
Inform user about data acquisition
Implicit data acquisition
Make recommendations instead of taking over actions
Taking over of work from the user
DIMINISHED PREDICTABILITY AND COMPREHENSIBILITY
1. Interface layout and responses
2. Success of adaptation
DIMINISHED CONTROLLABILITY
1. Specific behaviors
2. Content of user model
3. Overall mode of operation
OBTRUSIVENESS
INFRINGEMENT OF PRIVACY
DIMINISHED BREADTH OF EXPERIENCE
Allow inspection of user model
Explain system's actions
Submit actions to user for approval
Allow user to set parameters
Adapt timing of messages to context
Shift control to system gradually
Have user control data acquisition
Minimize accessibility of user data to others
Intentionally introduce diversity

FIGURE 19.21 Overview of usability challenges for user-adaptive systems and of ways of dealing with them. (Dashed arrows denote threats and solid arrows mitigation of threats, respectively; further explanation is given in the text.)

19.4.1 Threats to Predictability and Comprehensibility

The concept of *predictability* refers to the extent to which a user can predict the effects of his or her actions. *Comprehensibility* is the extent to which the user can understand system actions and/or has a clear picture of how the system works.* These goals are grouped together here because they are associated with largely the same set of other variables.

Users can try to predict and understand a system on two different levels of detail as follows:

1. *Exact layout and responses*. Especially detailed predictability is important when interface elements are involved that are accessed frequently by skilled users—for example, icons in control panels or options in menus (cf. the discussion of interface adaptation in Sections 19.2.2–3). In particular, the extreme case of predictability—remaining identical over time—has the advantage that after gaining experience users may be able to engage in *automatic processing* (see, e.g., Hammond [1987]; or, for a less academic discussion, Krug [2006]): They can use the parts of the interface quickly, accurately, and with little or no attention. In this situation, even minor deviations from constancy on a fine-grained level can have the serious consequence of making automatic processing impossible or error-prone. But even a lower degree of predictability on this detailed level can be useful for the user's planning of actions. Suppose that a person who regularly visits the website for this year's CHI conference knows that, if he or she types "chi" into the search field of his or her browser, the conference's home page will appear among the first few search results (possibly because the search is personalized and he or she has visited the conference page in the past): This knowledge will enable him or her to access the page more quickly than if the search engine's results were less predictable.
2. *Success of adaptation*. Often, all the user really needs to be able to predict and understand is the general level of success of the system's adaptation. For example, before spending time following up on a system's recommendations, the user may want to know how likely they are to be accurate. And if they turn out to be inaccurate, the user may want to understand why they were not satisfactory in this particular case, so as to be able to judge whether it will be worthwhile to consult the recommendations in the future.

19.4.2 Threats to Controllability

Controllability refers to the extent to which the user can bring about or prevent particular actions or states of the system if the user has the goal of doing so. It is an especially important issue if the system's adaptation consists of actions that have significant consequences, such as changing the UI or sending messages to other people. A widely used way of avoiding controllability problems is simply to have the system make recommendations, leaving it up to the user to take the actions in question. Or the system can take an action after the user has been asked to approve it. Both of these tactics can raise a threat of *obtrusiveness* (see Section 19.4.3); so it is important to find a way of making recommendations or asking for approval in an unobtrusive fashion but still in a noticeable fashion (see, e.g., the discussion of Ambient Help in Section 19.3.1.3).

Like predictability and comprehensibility, controllability can be achieved on various levels of granularity. Especially since the enhancement of controllability can come at a price, it is important to consider what kinds of control will really be desired. For example, there may be little point in submitting individual actions to the user for approval if the user lacks the knowledge or interest required to make the decisions. Jameson and Schwarzkopf (2002) found that users sometimes differ strikingly in their desire for control over a given aspect of adaptation, because they attach different weight to the advantages and disadvantages of controllability, some of which are situation-specific. This observation corroborates the recommendation of Wexelblat and Maes (1997) to make available several alternative types of control for users to choose from.

19.4.3 Obtrusiveness

We will use the term *obtrusiveness* to refer to the extent to which the system places demands on the user's attention, which reduces the user's ability to concentrate on his or her primary tasks. This term—and the related words *distracting* and *irritating*—were often heard in connection with early user-adaptive systems that were designed with inadequate attention to this possible side effect. Figure 19.21 shows that (1) there are several different reasons why user-adaptive systems may be obtrusive and (2) there are equally many strategies for minimizing obtrusiveness.

19.4.4 Threats to Privacy

Until a few years ago, threats to privacy were associated with user-adaptive systems more than with other types of systems, because adaptation implied a greater need to collect and store data about individual users (see, e.g., Cranor [2004]). Nowadays, where so much of everyday life has moved to the web, people have many reasons for storing personally sensitive information (including, e.g., their e-mail, personal photos, and work documents) on computers over which they have no direct control. So, the threat of privacy and security violations due to unauthorized access to or inappropriate use of personal data is now less strongly associated with the modeling of individual users. A comprehensive general discussion of privacy issues in human–computer interaction has been provided by Iachello and Hong (2007).

* The term *transparency* is sometimes used for this concept, but it can be confusing, because it also has a different, incompatible meaning.

A privacy threat that is still specifically associated with user-adaptive systems concerns the visibility of adaptation. For example, consider a reader of GOOGLE NEWS who suffers from a particular disease and has been reading news stories related to it. If the user is not eager for everyone to know about his or her disease, he or she may take care not to be seen reading such news stories when other people are present. But if he or she visits the personalized section of the news site when someone else is looking and a story about the disease appears there unexpectedly, the observer may be able to infer that the user is interested in the topic: The stories that are displayed implicitly reflect the content of the user model that the system has acquired. As Figure 19.21 indicates, a preventive measure is to give the user ways of limiting the visibility of potentially sensitive adaptation.

19.4.5 Diminished Breadth of Experience

When a user-adaptive system helps the user with some form of information acquisition (cf. Section 19.3), much of the work of examining the individual documents, products, and/or people involved is typically taken over by the system. A consequence can be that the user ends up learning less about the domain in question than the user would with a nonadaptive system (cf. Lanier [1995] for an early discussion of this issue).

Findlater and McGrenere (2010) investigated this type of trade-off in depth in connection with personalized UIs that limit the number of features that a user is exposed to. Their results confirmed that this type of personalization can both increase users' performance on their main tasks and reduce their awareness of features that might be useful with other tasks. The authors discuss a number of considerations that need to be taken into account when this type of trade-off is encountered.

As Figure 19.21 indicates, a general preventive measure is to ensure that users are free to explore the domain in question freely despite the adaptive support that the system offers. For example, recommender systems in e-commerce do not in general prevent the user from browsing or searching in product catalogs.

If a user does choose to rely heavily on the system's adaptations or recommendations, reduction of the breadth of experience is especially likely if the system relies on an incomplete user model (e.g., knowing about only a couple of the tasks that the user regularly performs or a couple of topics that the user is interested in). Some systems mitigate this problem by systematically proposing solutions that are *not* dictated by the current user model (see, e.g., Ziegler et al. [2005] for a method that is directly applicable to recommendation lists such as those of NETFLIX; and Linden, Hanks, and Lesh [1997] and Shearin and Lieberman [2001], for methods realized in different types of recommenders).

19.4.6 The Temporal Dimension of Usability Side Effects

The ways in which a user experiences a particular usability side effect with a given adaptive system can evolve as the user gains experience with the system. For example, adaptations that initially seem unpredictable and incomprehensible may become less so once the user has experienced them for a while. And a user may be able to learn over time how to control adaptations. In some cases, therefore, usability side effects represent an initial obstacle rather than a permanent drawback. On the other hand, since an initial obstacle may prompt the user to reject the adaptive functionality, it is worthwhile even in these cases to consider what can be done to improve the user's early experience. The remedial measure shown in Figure 19.21 of enabling the user to control the system closely at first and shift control to the system gradually is an example of such a strategy.

In general, though, the temporal evolution of the usability of an adaptive system is more complex than with nonadaptive systems, because the system tends to evolve even as the user is learning about it. A systematic way of thinking about the complex patterns that can result is offered by Jameson (2009).

19.5 OBTAINING INFORMATION ABOUT USERS

Any form of adaptation to an individual user presupposes that the system can acquire information about that user. Indeed, one reason for the recent increase in the prevalence of user-adaptive systems is the growth in possibilities for acquiring and exploiting such data.

Sections 19.5.1 and 19.5.2 will look, respectively, at (1) information that the user supplies to the system explicitly for the purpose of allowing the system to adapt and (2) information that the system obtains in some other way.

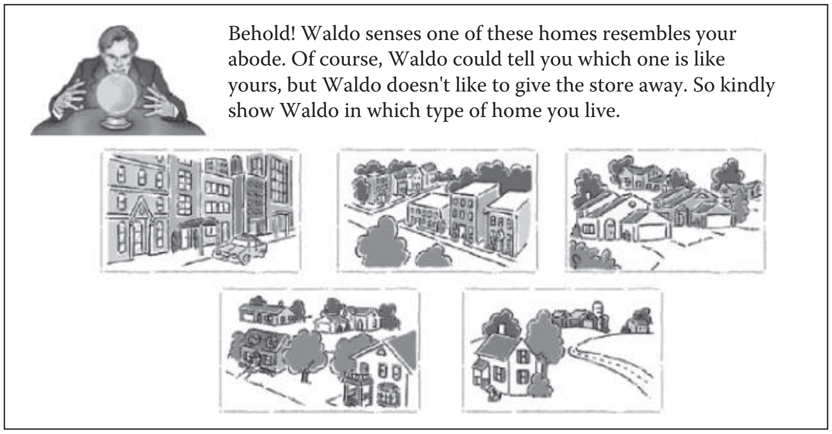
19.5.1 Explicit Self-Reports and -Assessments

19.5.1.1 Self-Reports about Objective Personal Characteristics

Information about objective properties of the user (such as age, profession, and place of residence) sometimes has implications that are relevant for system adaptation—for example, concerning the topics that the user is likely to be knowledgeable about or interested in. This type of information has the advantage of changing relatively infrequently. Some user-adaptive systems request information of this type from users, but the following caveats apply:

1. Specifying information such as profession and place of residence may require a fair amount of tedious menu selection and/or typing.
2. Since information of this sort can often be used to determine the user's identity, a user may justifiably be concerned about privacy. Even in cases where such concerns are unfounded, they may discourage the user from entering the requested information.

A general approach is to (1) restrict requests for personal data to the few pieces of information (if any) that the system really requires; and (2) explain the uses to which the data will be put. A number of suggestions about how the use of personally identifying data can be minimized are given by Cranor



Behold! Waldo senses one of these homes resembles your abode. Of course, Waldo could tell you which one is like yours, but Waldo doesn't like to give the store away. So kindly show Waldo in which type of home you live.

FIGURE 19.22 Example of a screen with which the LifeStyle Finder elicits demographic information. (Figure 19.3 of “Lifestyle Finder: Intelligent user profiling using large-scale demographic data,” by Krulwich, B. 1997. *AI Mag* 18(2):37–45. Research conducted at the Center for Strategic Technology Research of Andersen Consulting [now Accenture Technology Labs].) (Copyright 1997 by the American Association for Artificial Intelligence. Adapted with permission.)

(2004). An especially creative early approach appeared in the web-based Lifestyle Finder prototype (Figure 19.22; Krulwich 1997), which was characterized by a playful style and an absence of requests for personally identifying information. Of the users surveyed, 93% agreed that the Lifestyle Finder’s questions did not invade their privacy.

19.5.1.2 Self-Assessments of Interests and Knowledge

It is sometimes helpful for a user-adaptive system to have an assessment of a property of the user that can be expressed naturally as a position on a particular general dimension: the level of the user’s interest in a particular topic, the level of his or her knowledge about it, or the importance that the user attaches to a particular evaluation criterion. Often an assessment is arrived at through inference on the basis of indirect evidence, as with the assessments of the user’s interest in news items in the personalized section of Google News. But it may be necessary or more efficient to ask the user for an explicit assessment. For example, shortly before this chapter went to press and after the discussion of Google News in Section 19.3.1 had been completed, Google News began providing a form (shown in Figure 19.23) on which users could specify their interests explicitly.

Because of the effort involved in this type of self-assessment and the fact that the assessments may quickly become obsolete, it is in general worthwhile to consider ways of minimizing such requests, making responses optional, ensuring that the purpose is clear, and integrating the self-assessment process into the user’s main task (see, e.g., Tsandilas and Schraefel [2004], for some innovative ideas about how to achieve these goals).

19.5.1.3 Self-Reports on Specific Evaluations

Instead of asking a user to describe his or her interests explicitly, some systems try to infer the user’s position on the basis of his or her explicitly evaluative responses to specific items. Familiar examples include rating scales on which a user can award one to five stars and the now-ubiquitous thumbs-up “like” icon of Facebook. The items that the user evaluates can be (1) items that the user is currently experiencing directly (e.g., the current web page); (2) actions that the system has just performed, which the user may want to encourage or discourage (see, e.g., Wolfman et al. [2001]); (3) items that the user must judge on the basis of a description (e.g., the abstract of a talk; a table listing the attributes of a physical product); or (4) the mere name of an item (e.g., a movie) that the user may have had some experience with in the past. The cognitive effort required depends in part on how directly available the item is: In the third and fourth cases just listed, the user may need to perform memory retrieval and/or inference to arrive at an evaluation.

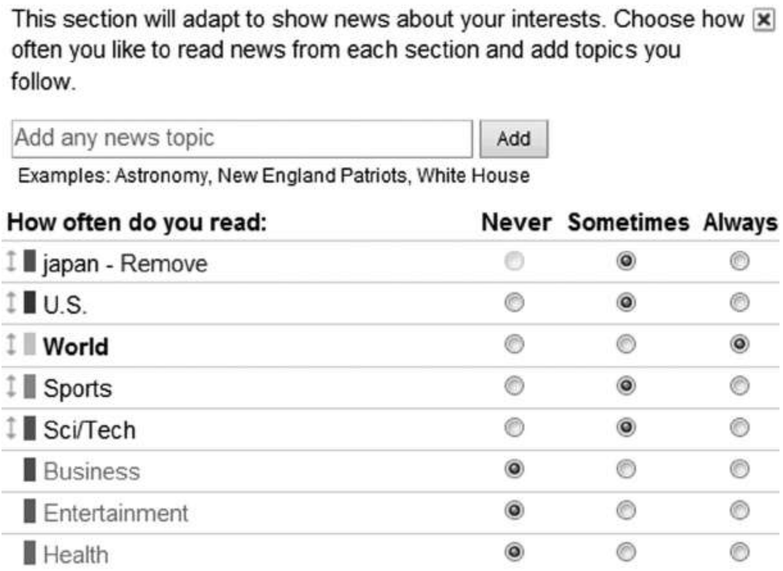
19.5.1.4 Responses to Test Items

In systems that support learning, it is often natural to administer tests of knowledge or skill. In addition to serving their normal educational functions, these tests can yield valuable information for the system’s adaptation to the user. An advantage of tests is that they can be constructed, administered, and interpreted with the help of a large body of theory, methodology, and practical experience (see, e.g., Wainer [2000]).

Outside of a learning context, users are likely to hesitate to invest time in tests of knowledge or skill unless these can be presented in an enjoyable form (see, e.g., the color discrimination test used by Gutkauf et al. [1997] to identify perceptual limitations relevant to the automatic generation of graphs). Trewin (2004) reports on experience with a brief typing test that was designed to identify helpful keyboard adaptations: Some users who turned out to require no adaptations were disappointed that their investment in the test had yielded no benefit. As a result, Trewin decided that adaptations should be based on the users’ naturally occurring typing behavior.

19.5.2 Nonexplicit Input

Section 19.5.1 has given some examples of why designers often look for ways of obtaining information about the user that does not require any explicit input by the user.



This section will adapt to show news about your interests. Choose how often you like to read news from each section and add topics you follow.

Add any news topic | Add

Examples: Astronomy, New England Patriots, White House

How often do you read:	Never	Sometimes	Always
japan - Remove	○	●	○
U.S.	○	●	○
World	○	○	●
Sports	○	●	○
Sci/Tech	○	●	○
Business	●	○	○
Entertainment	●	○	○
Health	●	○	○

FIGURE 19.23 A form in which a user of GOOGLE NEWS can characterize her interests in particular types of news.

19.5.2.1 Naturally Occurring Actions

The broadest and most important category of information of this type includes all of the actions that the user performs with the system that do not have the purpose of revealing information about the user to the system. These may range from major actions like purchasing an expensive product to minor ones like scrolling down a web page. The more significant actions tend to be specific to the particular type of system that is involved (e.g., e-commerce sites vs. learning environments).

In their pure form, naturally occurring actions require no additional investment by the user. The main limitation is that they are hard to interpret; for example, the fact that a given web page has been displayed in the user's browser for 4 minvutes does not reveal with certainty which (if any) of the text displayed on that page the user has actually read. Some designers have tried to deal with this trade-off by designing the UI in such a way that the naturally occurring actions are especially easy to interpret. For example, a web-based system might display just one news story on each page, even if displaying several stories on each page would normally be more desirable.

19.5.2.2 Information from Social Networks

One type of information about users that has grown explosively during the last several years is information that can be found in the increasingly ubiquitous social networks (e.g., FACEBOOK, LINKEDIN, and ORKUT, but also media-sharing sites such as FLICKR). Much of this information is similar in nature to information that can in principle be found elsewhere—for example, on a user's personal home page or in their e-mail messages—but social networking sites encourage people to create and expose more of this information than they otherwise would. One type of information is specific to social networks: explicit links connecting people (e.g., as "friends," professional collaborators, or members of the same online community). The most obvious way of exploiting link information was illustrated by the SOCIALBLUE system (Section 19.3.4): helping people to create additional links of the same types. But the fact that a given user is a friend of another person or a member of a given community can enable the system to make many other types of inference about that user by examining the persons to whom he or she is linked (see, e.g., Brzozowski, Hogg, and Szabo [2008]; Mislove et al. [2010]; Schifanella et al. [2010]; Zheleva and Getoor [2009]). In effect, much of the information that can be acquired in other ways summarized in this section can be propagated to other users via such links—although the nature of the inferences that can be made depends on the nature of the links and the type of information that is involved.

19.5.2.3 Other Types of Previously Stored Information

Even before the advent of social networking platforms, there were ways in which some user-adaptive systems could access relevant information about a user which was acquired and stored independently of the system's interaction with the user:

1. If the user has some relationship (e.g., patient, customer) with the organization that operates the system, this organization may have information about the user that it has stored for reasons unrelated to any adaptation, such as the user's medical record (see Cawsey et al. [2007] for examples) or address.
2. If there is some other system that has already built up a model of the user, the system may be able to access the results of that modeling effort and try to apply them to its own modeling task. There is a line of research that deals with *user modeling servers* (see, e.g., Kobsa [2007]): systems that store information about users centrally and supply such information to a number of different applications.

Related concepts are *ubiquitous user modeling* (see, e.g., Heckmann [2005]) and *cross-system personalization* (Mehta 2009).

19.5.2.4 Low-Level Indices of Psychological States

The next two categories of information about the user started to become practically feasible in the late 1990s with advances in the miniaturization of sensing devices.

The first category of sensor-based information (discussed at length in the classic book of Picard [1997]) comprises data that reflect aspects of a user's psychological state.

Two categories of sensing devices have been used: (1) devices attached to the user's body (or to the computing device itself) that transmit physiological data, such as electromyogram signals, the galvanic skin response, blood volume pressure, and the pattern of respiration (see Lisetti and Nasoz [2004], for an overview); and (2) video cameras and microphones that transmit psychologically relevant information about the user, such as features of his or her facial expressions (see, e.g., Bartlett et al. [2003]), speech (see, e.g., Liscombe, Riccardi, and Hakkani-Tür [2005]), or eye movements (see, e.g., Conati and Merten [2007]).

With both categories of sensors, the extraction of meaningful features from the low-level data stream requires the application of pattern recognition techniques. These typically make use of the results of machine learning studies in which relationships between low-level data and meaningful features have been learned.

One advantage of sensors is that they supply a continuous stream of data, the cost to the user being limited to the physical and social discomfort that may be associated with the carrying or wearing of the devices. These factors have been diminishing steadily in importance over the years with advances in miniaturization.

19.5.2.5 Signals Concerning the Current Surroundings

As computing devices become more portable, it is becoming increasingly important for a user-adaptive system to have information about the user's current surroundings. Here again, two broad categories of input devices can be distinguished (see Krüger et al. [2007], for a discussion of a number of specific types of devices).

1. Devices that receive explicit signals about the user's surroundings from specialized transmitters. The use of GPS (global positioning system) technology, often in conjunction with other signals, to determine a user's current location is familiar to most users of modern smartphones, and one of the purposes is to personalize the provision of information (e.g., about local attractions). More specialized transmitters and receivers are required, for example, if a portable museum guide is to be able to determine which exhibit the user is looking at.
2. More general sensing or input devices. For example, Schiele et al. (2001) describe the use of a miniature video camera and microphone (each roughly the size of a coin) that enable a wearable computer to discriminate among different types of surroundings (e.g., a supermarket vs. a street). The use of general-purpose sensors eliminates the dependence on specialized transmitters. On the other hand, the interpretation of the signals requires the use of sophisticated machine learning and pattern recognition techniques.

19.6 CONCLUDING REFLECTIONS

During the past few years, an increasing range of systems have been put into widespread use that exhibit some form of adaptation to users; in Section 19.2–3 have presented a representative sample. This increasing pervasiveness can be explained in part in terms of advances that have increased the feasibility of successful adaptation to users: better ways of acquiring and processing relevant information about users and increases in computational capacity for realizing the adaptation. But there has also been a growth in understanding of the forms of adaptation that fit with the ways in which people like to use computing technology, providing added value while avoiding the potential usability side effects discussed in Section 19.4.

One general design pattern has emerged which has been applied successfully in various forms and which might be considered the default design pattern to consider for any new form of adaptation: the nonadaptive interaction with an application is supplemented with recommendations that the user can optionally consider and follow up on.

The earliest widely used examples of this general pattern included product recommenders for e-commerce, such as Amazon.com's recommendations. As was illustrated by the examples in Sections 19.2–3, the pattern has also been appearing with other functions of adaptation, such as personalized news, recommendations of people to link up with, and support for the discovery and learning of useful commands in a complex application. In tutoring systems that include an "outer loop," recommendations can concern the suggestions of learning material and exercises. Even some forms of adaptation that would not normally be called "recommendation," such as split interfaces and TaskTracer's support for the performance of routine tasks shown in Figure 19.4, fit the same basic pattern.

The general appeal of this design pattern is understandable in that it involves making available to users some potentially helpful options which they would have had some difficulty in identifying themselves or which at least would have taken some time for them to access. This benefit is provided with little or no cost in terms of usability side effects: provided that the available display space is adequate, the additional options can be offered in an unobtrusive way. The fact that the user is free to choose what to do with the recommended options—or to ignore them—means that any difficulty in predicting or understanding them need not cause significant problems; that the system does not take any significant action that is beyond the user's control; and that the user's experience does not have to be restricted.

This relatively straightforward and generally successful paradigm cannot be applied to all forms of adaptation to users. Adaptation to abilities and impairments often requires the provision of an alternative interface (cf. Section 19.2.4). And some types of system—such as small mobile devices, smart objects embedded in the environment, and telephone-based spoken dialog systems—may lack sufficient display space to offer additional options unobtrusively or a convenient way for users to select such options. Achieving effective and widely used adaptation where the general recommendation-based design pattern cannot be applied remains a challenge for researchers and designers.

ACKNOWLEDGMENTS

The following colleagues provided valuable assistance in the form of comments on earlier drafts, supplementary information about their research, and/or graphics depicting their systems: Jilin Chen, Thomas Dietterich, Werner Geyer, Shaun Kane, Bruce Krulwich, Justin Matejka, and Bruce McLaren. The contribution of the first author was supported in part by the 7th Framework EU Integrating Project GLOCAL: *Event-based Retrieval of Networked Media* (http://www.glocal-project.eu/) under grant agreement 248984.

REFERENCES

Aleven, V., B. M. McLaren, J. Sewall, and K. R. Koedinger. 2009. A new paradigm for intelligent tutoring systems: Example-tracing tutors. *Int J Artif Intell Educ* 19(2):105–54.

Barnard, L., J. Yi, J. Jacko, and A. Sears. 2007. Capturing the effects of context on human performance in mobile computing systems. *Pers Ubiquitous Comput* 11(2):81–96.

Bartlett, M. S., G. Littlewort, I. Fasel, and J. R. Movellan. 2003. Real time face detection and facial expression recognition: Development and applications to human computer interaction. In *Proceedings of the Workshop on Computer Vision and Pattern Recognition for Human-Computer Interaction at the 2003 Conference on Computer Vision and Pattern Recognition*, 53–8. New York: IEEE.

Beales, H. 2010. *The Value of Behavioral Targeting.* Study sponsored by the Network Advertising Initiative. http://www. net work-advertising.org/pdfs/Beales_NAI_Study.pdf. Accessed date: October 1st, 2011.

Bergman, E., and E. Johnson. 1995. Towards accessible human-computer interaction. *Adv Hum Comput Interact* 5(1). In *Advances in Human-Computer Interaction*, ed. J. Nielsen, 87–114. Norwood, NJ: Ablex.

Billsus, D., D. Hilbert, and D. Maynes-Aminzade. 2005. Improving proactive information systems. In *IUI 2005: International Conference on Intelligent User Interfaces,* ed. J. Riedl, A. Jameson, D. Billsus, and T. Lau, 159–66. New York: ACM.

Billsus, D., and M. J. Pazzani. 2007. Adaptive news access. In *The Adaptive Web: Methods and Strategies of Web Personalization,* ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 550–72. Berlin, Germany: Springer.

Bontcheva, K., and Y. Wilks. 2005. Tailoring automatically generated hypertext. *User Model User-adapt Interact* 15:135–68.

Brettel, H., F. Viénot, and J. D. Mollon. 1997. Computerized simulation of color appearance for dichromats. *J Opt Soc Am A* 14(10):2647–55.

Brusilovsky, P., A. Kobsa, and W. Nejdl, eds. 2007. *The Adaptive Web: Methods and Strategies of Web Personalization.* Berlin, Germany: Springer.

Brzozowski, M. J., T. Hogg, and G. Szabo. 2008. Friends and foes: Ideological social networking. In *Human Factors in Computing Systems: CHI 2008 Conference Proceedings*, ed. M. Burnett, M. F. Costabile, T. Catarci, B. de Ruyter, D. Tan, M. Czerwinski, and A. Lund, 817–20. New York: ACM.

Budzik, J., K. Hammond, and L. Birnbaum. 2001. Information access in context. *Knowl Based Syst* 14:37–53.

Bunt, A., G. Carenini, and C. Conati. 2007. Adaptive content presentation for the web. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 409–32. Berlin, Germany: Springer.

Bunt, A., C. Conati, and J. McGrenere. 2007. Supporting interface customization using a mixed-initiative approach. In *IUI 2007: International Conference on Intelligent User Interfaces*, ed. T. Lau and A. R. Puerta. New York: ACM.

Burke, R. D., K. J. Hammond, and B. C. Young. 1997. The FindMe approach to assisted browsing. *IEEE Expert* 12(4):32–40.

Canadian Marketing Association 2009. Behavioural advertising: Why consumer choice matters. Published on-line as part of the CMA Leadership Series at http://www.the-cma.org/marketingresources/downloads/2009BehaviouralAdvertising.pdf. Latest access: October 1st, 2011.

Carmagnola, F., F. Vernero, and P. Grillo. 2009. SoNARS: A social networks-based algorithm for social recommender systems. In *Proceedings of UMAP 2009, the 17th international Conference on User Modeling, Adaptation, and Personalization*, 223–234. Berlin, Springer

Cawsey, A., F. Grasso, and C. Paris. 2007. Adaptive information for consumers of healthcare. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 465–84. Berlin, Germany: Springer.

Chen, J., W. Geyer, C. Dugan, M. Muller, and I. Guy. 2009. “Make new friends, but keep the old”—Recommending people on social networking sites. In *Human Factors in Computing Systems: CHI 2009 Conference Proceedings*, ed. S. Greenberg, S. Hudson, K. Hinckley, M. R. Morris, and D. R. Olsen, 201–10. New York: ACM.

Chen, W., D. Zhang, and E. Y. Chang. 2008. Combinational collaborative filtering for personalized community recommendation. In *Proceedings of the Fourteenth International Conference on Knowledge Discovery and Data Mining*, 115–123. New York: ACM.

Conati, C., and C. Merten. 2007. Eye-tracking for user modeling in exploratory learning environments: An empirical evaluation. *Knowl Based Syst* 20(6):557–74.

Cranor, L. F. 2004. ‘I didn’t buy it for myself’: Privacy and ecommerce personalization. In *Designing Personalized User Experiences in Ecommerce,* ed. C. Karat, J. Blom, and J. Karat, 57–74. Dordrecht, The Netherlands: Kluwer.

Daly, E. M., W. Geyer, and D. R. Millen. 2010. The network effects of recommending social connections. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, ed. P. Resnick, M. Zanker, X. Amatriain, and M. Torrens, 301–4. New York: ACM.

Dietterich, T. G., X. Bao, V. Keiser, and J. Shen. 2010. Machine learning methods for high level cyber situation awareness. In *Cyber Situational Awareness: Issues and Research*, ed. S. Jajodia, P. Liu, V. Swarup, and C. Wang. New York: Springer.

Ehrlich, K., C. Lin, and V. Griffiths-Fisher. 2007. Searching for experts in the enterprise: Combining text and social network analysis. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work*, 117–26. New York: ACM.

Federal Trade Commission, T. 2009. *Federal Trade Commission Staff Report: Self-Regulatory Principles for Online Behavioral Advertising.* http://www.ftc.gov/os/2009/02/P085400behavad report.pdf. Latest access: October 1st, 2011.

Findlater, L., and K. Z. Gajos. 2009. Design space and evaluation challenges of adaptive graphical user interfaces. *AI Mag* 30(4):68–73.

Findlater, L., and J. McGrenere. 2008. Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces. In *Human Factors in Computing Systems: CHI 2008 Conference Proceedings*, ed. M. Burnett, M. F. Costabile, T. Catarci, B. de Ruyter, D. Tan, M. Czerwinski, and A. Lund, 1247–56. New York: ACM.

Findlater, L., and J. McGrenere. 2010. Beyond performance: Feature awareness in personalized interfaces. *Int J Hum Comput Stud* 68(3):121–37.

Findlater, L., K. Moffatt, J. Mc Grenere, and J. Dawson. 2009. Ephemeral adaptation: The use of gradual onset to improve menu selection performance. In *Human Factors in Computing Systems: CHI 2009 Conference Proceedings*, ed. S. Greenberg, S. Hudson, K. Hinckley, M. R. Morris, and D. R. Olsen., 1655–64 New York: ACM.

Gajos, K. Z., M. Czerwinski, D. S. Tan, and D. S. Weld. 2006. Exploring the design space for adaptive graphical user interfaces. In *Proceedings of the 2006 Conference on Advanced Visual Interfaces*, 201–8. New York: ACM.

Gajos, K. Z., D. S. Weld, and J. O. Wobbrock. 2010. Automatically personalized user interfaces with Supple. *Artif Intell* 174(12–13):910–50.

Gajos, K. Z., J. O. Wobbrock, and D. S. Weld. 2007. Automatically generating user interfaces adapted to users' motor and vision capabilities. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology,* 231–40. New York: ACM.

Gajos, K. Z., J. O. Wobbrock, and D. S. Weld. 2008. Improving the performance of motor-impaired users with automatically generated, ability-based interfaces. In *Human Factors in Computing Systems: CHI 2008 Conference Proceedings*, ed. M. Burnett, M. F. Costabile, T. Catarci, B. de Ruyter, D. Tan, M. Czerwinski, and A. Lund, 1257–66. New York: ACM.

Gervasio, M. T., M. D. Moffitt, M. E. Pollack, J. M. Taylor, and T. E. Uribe. 2005. Active preference learning for personalized calendar scheduling assistance. In *IUI 2005: International Conference on Intelligent User Interfaces,* ed. J. Riedl, A. Jameson, D. Billsus, and T. Lau, 90–7. New York: ACM.

Gutkauf, B., S. Thies, and G. Domik. 1997. A user-adaptive chart editing system based on user modeling and critiquing. In *User Modeling: Proceedings of the Sixth International Conference, UM97,* ed. A. Jameson, C. Paris, and C. Tasso, 159–70. Vienna: Springer Wien New York.

Hammond, N. 1987. Principles from the psychology of skill acquisition. In *Applying Cognitive Psychology to User-Interface Design,* ed. M. M. Gardiner and B. Christie, 163–88. Chichester, England: Wiley.

Heckmann, D. 2005. *Ubiquitous User Modeling.* Berlin: infix.

Hegner, S. J., P. McKevitt, P. Norvig, and R. L. Wilensky, eds. 2001. *Intelligent Help Systems for UNIX.* Dordrecht, The Netherlands: Kluwer.

Höök, K. 2000. Steps to take before IUIs become real. *Interact Comput* 12(4):409–26.

Horvitz, E. 1999. Principles of mixed-initiative user interfaces. In *Human Factors in Computing Systems: CHI 1999 Conference Proceedings,* ed. M. G. Williams, M. W. Altom, K. Ehrlich, and W. Newman, 159–66. New York: ACM.

Hwang, F., S. Keates, P. Langdon, and J. Clarkson. 2004. Mouse movements of motion-impaired users: A submovement analysis. In *Proceedings of the 6th International ACM SIGACCESS Conference on Computers and Accessibility,* 102–9. New York: ACM.

Iachello, G., and J. Hong. 2007. End-user privacy in human-computer interaction. *Found Trends Hum Comput Interact* 1(1):1–137.

Jameson, A. 2003. Adaptive interfaces and agents. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, ed. J. A. Jacko and A. Sears, 305–30. Mahwah, NJ: Erlbaum.

Jameson, A. 2008. Adaptive interfaces and agents. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications,* ed. A. Sears and J. A. Jacko, 2nd ed., 433–58. Boca Raton, FL: CRC Press.

Jameson, A. 2009. Understanding and dealing with usability side effects of intelligent processing. *AI Mag* 30(4):23–40.

Jameson, A., and E. Schwarzkopf. 2002. Pros and cons of controllability: An empirical study. In *Adaptive Hypermedia and Adaptive Web-Based Systems: Proceedings of AH 2002*, ed. P. De Bra, P. Brusilovsky, and R. Conejo, 193–202. Berlin, Germany: Springer.

Jameson, A., and B. Smyth. 2007. Recommendation to groups. In *The Adaptive Web: Methods and Strategies of Web Personalization,* ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 596–627. Berlin, Germany: Springer.

Jefferson, L., and R. Harvey. 2006. Accommodating color blind computer users. In *Proceedings of the Eighth International ACM SIGACCESS Conference on Computers and Accessibility*, 40–7. New York: ACM.

Jefferson, L., and R. Harvey. 2007. An interface to support color blind computer users. In *Human Factors in Computing Systems: CHI 2007 Conference Proceedings,* ed. B. Begole, S. Payne, E. Churchill, R. S. Amant, D. Gilmore, and M. B. Rosson, 1535–8. New York: ACM.

Kane, S. K., J. O. Wobbrock, and I. E. Smith. 2008. Getting off the treadmill: Evaluating walking user interfaces for mobile devices in public spaces. In *Proceedings of the 10th International Conference on Human Computer Interaction With Mobile Devices and Services*, 109–18. New York: ACM.

Keates, S., P. Langdon, J. P. Clarkson, and P. Robinson. 2002. User models and user physical capability. *User Model User-adapt Interact* 12(2):139–69.

Kelly, D., and J. Teevan. 2003. Implicit feedback for inferring user preference: A bibliography. *ACM SIGIR Forum* 37(2):18–28.

Kobsa, A. 2007. Generic user modeling systems. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 136–54. Berlin, Germany: Springer.

Krug, S. 2006. *Don't Make Me Think: A Common-Sense Approach to Web Usability*, 2nd ed. Berkeley, CA: New Riders.

Krüger, A., J. Baus, D. Heckmann, M. Kruppa, and R. Wasinger. 2007. Adaptive mobile guides. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 521–49. Berlin, Germany: Springer.

Krulwich, B. 1997. Lifestyle Finder: Intelligent user profiling using large-scale demographic data. *AI Mag* 18(2):37–45.

Lanier, J. 1995. Agents of alienation. *Interactions* 2(3):66–72.

Law, C. M., A. Sears, and K. J. Price. 2005. Issues in the categorization of disabilities for user testing. In *Proceedings of HCI International*. Mahwah, NJ: Lawrence Erlbaum Associates.

Li, W., J. Matejka, T. Grossman, J. Konstan, and G. Fitzmaurice. 2011. Design and evaluation of a command recommendation system for software applications. *ACM Trans Comput Hum Interact* 18(2), Article 6.

Lin, M., R. Goldman, K. J. Price, A. Sears, and J. Jacko. 2007. How do people tap when walking? An empirical investigation of nomadic data entry. *Int J Hum Comput Stud* 65(9):759–69.

Linden, G., S. Hanks, and N. Lesh. 1997. Interactive assessment of user preference models: The automated travel assistant. In *User Modeling: Proceedings of the Sixth International Conference, UM97*, ed. A. Jameson, C. Paris, and C. Tasso, 67–78. Vienna: Springer Wien New York.

Liscombe, J., G. Riccardi, and D. Hakkani-Tür. 2005. Using context to improve emotion detection in spoken dialog systems. In *Proceedings of the Ninth European Conference on Speech Communication and Technology*, 1845–8. Grenoble: ISCA.

Lisetti, C. L., and F. Nasoz. 2004. Using noninvasive wearable computers to recognize human emotions from physiological signals. *EURASIP J Appl Signal Process* 11:1672–87.

Liu, J., P. Dolan, and E. R. Pedersen. 2010. Personalized news recommendation based on click behavior. In *IUI 2010: International Conference on Intelligent User Interfaces,* ed. M. O. Cavazza and M. X. Zhou, 31–40. New York: ACM.

Mackay, W. E. 1991. Triggers and barriers to customizing software. In *Human Factors in Computing Systems: CHI 1991 Conference Proceedings*, ed. S. P. Robertson, G. M. Olson, and J. S. Olson, 153–60. New York: ACM.

Maes, P. 1994. Agents that reduce work and information overload. *Commun ACM* 37(7):30–40.

Matejka, J., W. Li, T. Grossman, and G. Fitzmaurice. 2009. Community Commands: Command recommendations for software applications. In *UIST 2009 Conference Proceedings: ACM Symposium on User Interface Software and Technology*, 193–202. New York: ACM.

Matejka, J., T. Grossman, and G. Fitzmaurice. 2011. Ambient Help. In *Human Factors in Computing Systems: CHI 2011 Conference Proceedings*, ed. D. Tan, B. Begole, W. Kellogg, G. Fitzpatrick, and C. Gutwin. New York: ACM.

McGrenere, J., R. M. Baecker, and K. S. Booth. 2007. A field evaluation of an adaptable two-interface design for feature-rich software. *Trans Comput Hum Interact* 14(1), Article 3.

McLaren, B. M., K. E. DeLeeuw, and R. E. Mayer. 2011a. A politeness effect in learning with web-based intelligent tutors. *Int J Hum Comput Stud* 69:70–9.

McLaren, B. M., K. E. DeLeeuw, and R. E. Mayer. 2011b. Polite web-based intelligent tutors: Can they improve learning in classrooms? *Comput Educ* 56:574–84.

Mehta, B. 2009. *Cross System Personalization: Enabling Personalization Across Multiple Systems*. Saarbrücken, Germany: VDM Verlag.

Mislove, A., B. Viswanath, P. K. Gummadi, and P. Druschel. 2010. You are who you know: Inferring user profiles in online social networks. In *Proceedings of the Third International Conference on Web Search and Web Data Mining*, 251–60. New York: ACM.

Mitchell, T., R. Caruana, D. Freitag, J. McDermott, and D. Zabowski. 1994. Experience with a learning personal assistant. *Commun ACM* 37(7):81–91.

Mitchell, J., and B. Shneiderman. 1989. Dynamic versus static menus: An exploratory comparison. *SIGCHI Bull* 20(4):33–7.

Norman, D. A. 1994. How might people interact with agents? *Commun ACM* 37(7):68–71.

Palen, L. 1999. Social, individual and technological issues for groupware calendar systems. In *Human Factors in Computing Systems: CHI 1999 Conference Proceedings*, ed. M. G. Williams, M. W. Altom, K. Ehrlich, and W. Newman, 17–24. New York: ACM.

Picard, R. W. 1997. *Affective Computing.* Cambridge, MA: MIT Press.

Pu, P., and L. Chen. 2008. User-involved preference elicitation for product search and recommender systems. *AI Mag* 29(4):93–103.

Rhodes, B. J. 2000. *Just-in-Time Information Retrieval.* Dissertation, School of Architecture and Planning, Massachusetts Institute of Technology. Available from http://www.bradleyrhodes.com/Papers/rhodes-phd-JITIR.pdf; latest access: October 1st, 2011.

Rich, C. 2009. Building task-based user interfaces with ANSI/CEA-2018. *IEEE Comput* 42(8):20–7.

Rich, C., C. Sidner, N. Lesh, A. Garland, S. Booth, and M. Chimani. 2005. DiamondHelp: A collaborative interface framework for networked home appliances. In *5th International Workshop on Smart Appliances and Wearable Computing, IEEE International Conference on Distributed Computing Systems Workshops*, 514–9. New York: IEEE.

Schafer, J. B., D. Frankowski, J. Herlocker, and S. Sen. 2007. Collaborative filtering recommender systems. In *The Adaptive Web: Methods and Strategies of Web Personalization*, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 291–324. Berlin, Germany: Springer.

Schiele, B., T. Starner, B. Rhodes, B. Clarkson, and A. Pentland. 2001. Situation aware computing with wearable computers. In *Fundamentals of Wearable Computers and Augmented Reality*, ed. W. Barfield and T. Caudell, 511–38. Mahwah, NJ: Erlbaum.

Schifanella, R., A. Barrat, C. Cattuto, B. Markines, and F. Menczer. 2010. Folks in folksonomies: Social link prediction from shared metadata. In *Proceedings of the Third International Conference on Web Search and Web Data Mining*, 271–80. New York: ACM.

Shami, N. S., Y. C. Yuan, D. Cosley, L. Xia, and G. Gay. 2007. That's what friends are for: Facilitating 'who knows what' across group boundaries. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work*, 379–82.

Shearin, S., and H. Lieberman. 2001. Intelligent profiling by example. In *IUI 2001: International Conference on Intelligent User Interfaces*, ed. J. Lester, 145–51. New York: ACM.

Shen, J., E. Fitzhenry, and T. Dietterich. 2009. Discovering frequent work procedures from resource connections. In *IUI 2009: International Conference on Intelligent User Interfaces*, ed. N. Oliver and D. Weld. New York: ACM.

Shen, J., J. Irvine, X. Bao, M. Goodman, S. Kolibab, A. Tran, F. Carl, B. Kirschner, S. Stumpf, and T. Dietterich. 2009. Detecting and correcting user activity switches: Algorithms and interfaces. In *IUI 2009: International Conference on Intelligent User Interfaces*, ed. N. Oliver and D. Weld. New York: ACM.

Soller, A. 2007. Adaptive support for distributed collaboration. In *The Adaptive Web: Methods and Strategies of Web Personalization,* ed. P. Brusilovsky, A. Kobsa, and W. Nejdl, 573–95. Berlin, Germany: Springer.

Teevan, J., S. T. Dumais, and E. Horvitz. 2010. Potential for personalization. *ACM Trans Comput Hum Interact* 17(1).

Terveen, L., and D. W. McDonald. 2005. Social matching: A framework and research agenda. *ACM Trans Comput Hum Interact* 12(3):401–34.

Tintarev, N., and J. Masthoff. 2010. Explanation of recommendations. In *Recommender Systems Handbook*, ed. F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. Berlin, Germany: Springer.

Trewin, S. 2004. Automating accessibility: The Dynamic Keyboard. In *Proceedings of ASSETS 2004*, 71–8. New York: ACM.

Tsandilas, T., and M. Schraefel. 2004. Usable adaptive hypermedia. *New Rev Hypermedia Multimedia* 10(1):5–29.

VanLehn, K. 2006. The behavior of tutoring systems. *Int J Artif Intell Educ* 16(3):227–65.

Vasuki, V., N. Natarajan, Z. Lu, and I. S. Dhillon. 2010. Affiliation recommendation using auxiliary networks. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, ed. P. Resnick, M. Zanker, X. Amatriain, and M. Torrens, 103–10. New York: ACM.

Vig, J., S. Sen, and J. Riedl. 2009. Tagsplanations: Explaining recommendations using tags. In *IUI 2009: International Conference on Intelligent User Interfaces*, ed. N. Oliver and D. Weld, 47–56. New York: ACM.

Wainer, H., ed. 2000. *Computerized Adaptive Testing: A Primer.* Hillsdale, NJ: Erlbaum.

Wexelblat, A., and P. Maes. 1997. *Issues for Software Agent UI.* Unpublished manuscript, cited with permission. Available from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.8886. Latest access: October 1st, 2011.

Wobbrock, J. O., S. K. Kane, K. Z. Gajos, S. Harada, and J. Froelich. 2011. Ability-based design: Concept, principles and examples. *ACM Trans Access Comput*., 3(3), Article 9.

Wolfman, S. A., T. Lau, P. Domingos, and D. S. Weld. 2001. Mixed initiative interfaces for learning tasks: SMARTedit talks back. In *IUI 2001: International Conference on Intelligent User Interfaces,* ed. J. Lester, 167–74. New York: ACM.

Zheleva, E., and L. Getoor. 2009. To join or not to join: The illusion of privacy in social networks with mixed public and private user profiles. In *Proceedings of the 2009 International World Wide Web Conference*, 531–40. Published by the International World Wide Web Conference Committee, http://www.iw3c2.org/.

Ziegler, C., S. M. McNee, J. A. Konstan, and G. Lausen. 2005. Improving recommendation lists through topic diversification. In *Proceedings of the 2003 International World Wide Web Conference*, 22–32. Published by the International World Wide Web Conference Committee, http://www.iw3c2.org/.

20 Mobile Interaction Design in the Age of Experience Ecosystems

Marco Susani

CONTENTS

20.1 MOBILE EXPERIENCE

20.1.1 Interacting with Networked Ecosystems

The parallel evolution of network technologies and social behaviors is transforming the character of interactive systems.

The conventional model of the individual working in front of a personal computer is challenged by innovative ways of accessing content and communication: networked mobile wireless devices grow their functionality, handheld interfaces become richer, mobile-distributed delivery of content will be fully integrated with conventional media, and entirely new social forms of communication are emerging.

The mobile component of this system of interactions, centered on "the device formerly known as the cell phone," has been among the most impressive technocultural phenomena of the past 10 years, and its evolution is still changing drastically the way human interaction is conceived. Mobility is only one aspect of this revolution: mobile devices also happen to be much more related to the individuals who use them and follow constantly both their private and work life. Thus, the interaction with these tools is characterized by a combination of functional and emotional aspects.

Present interaction models, even the ones developed respecting the rules of usability, often lack fundamental emotional qualities that are needed to support and increase existing social behaviors and rituals.

The paradigm of pervasive and ubiquitous networking, instead of gravitating around the personal computer, is taking the shape of an ecosystem of mobile and static platforms, for individual and collective use, all connected to the network, and their coordination is taking the shape of a choreography of interactions to provide access to integrated content and communication.

"Convergence" seems finally to take shape, centered on three platforms that dominate this experience ecosystem:

- Virtually all personal computers are on the net, and most of them can access the net via mobile wireless technologies. Some of them, netbooks, become lean clients and access both documents and apps from the "cloud" (internet-based servers).

- Tablets fill the space between mobile ‘pocketable’ devices, such as phones, and ‘static’ devices, like computers. Tablets open an enormous opportunity to interact with digital content both at home and while mobile, and evolve the touch-based interaction paradigm and high quality graphics to a higher level.
- Almost 5 billion cell phones and smartphones, most of them capable to network both voice and net access, are in the hands of people of all cultures and statuses.
- The transition from cable and broadcast TV to Internet Protocol (IP) networked TV is transforming the paradigm of interacting with video, movies, and TV content. Around the TV, other distributed devices introduce new sizes and form factors to increase the mobility within the home.
- Embedded networking capabilities are growing in other “appliances,” such as cars, or in any “thing” (via the radio frequency identification [RFID] technology), and in active environments.
- All the above is going to be connected with a multiplicity of technologies; the integration of wired and wireless, and the integration of different technologies within each one of these domains (e.g., cellular networks talking to local wireless networks at different scales, from Bluetooth to Wi-Fi to RFID), transforms the picture of a network into a flexible, capable, seamless, “opportunistic” infrastructure that can take advantage of any available connectivity.

Similar social and cultural disruptions are happening in parallel with this fully developed technology disruption:

- Format and genres of communication integrate and hybridize, and social networking becomes mainstream.
- Content becomes deconstructed and pervasive.
- Based on the intersection of the two aspects above, new types of communities are emerging, and new social models growing around the generation and distribution of content explore the space between one-to-one communication (the evolution of the telephone) and broadcast (the evolution of mass media).

Devices of different types, sizes, and context of use become multiple points of access to the network, and the net is much more embedded in the everyday reality, both because it is more integrated with spaces and artifacts of everyday life and because it influences more and more human activities and social systems.

20.1.2 Content, Communication, and Control

The original interaction with personal computers focused on doing (writing a document, making calculations on a spread sheet, etc.), and its physical context was a single individual user sitting in front of his computer, isolated from the surrounding physical space. Today, the merging of computer technologies with telecommunication technologies and wireless network access calls for a completely different paradigm: computers, mobile devices, and networks are a complex set of tools not only to ‘do’ things to performing tasks. Instead, they are above all gates to communication media that connect people, hybrid physical and digital collaborative spaces to meet others, and tools to access published, dynamic, and ubiquitous content.

The interfaces of these devices are no longer exclusively mediators between a machine and an individual but much more often they are the interfaces that structure the user’s communication mental model or interfaces that organize, contextualize, and categorize in an accessible way the vast space of networked content.

Simplicity, the key attribute that allowed in the past easy access to the functionality of a “machine,” becomes today a much more sophisticated concept, because a “simple” access to the user’s sphere of communication and content cannot come only from a functional, efficient organization of the interaction, but is much more rooted in subjective and cultural aspects, such as the ability to handle speed and quantity of connections or the mental construct of the user’s social universe. The success of sociotechnical phenomena like Twitter, for example, demonstrates that communication at a pace that would be defined as overwhelming by the average adult is totally acceptable for younger generations.

In this chapter, we will investigate the specifics of interaction with a multiplicity of mobile networked devices, and we will see how the whole paradigm of networked reality changes the way we think about human–computer interaction (HCI). We will look at any networked device, whether monofunctional or multifunctional, such as cell phones, networked photo-cameras, networked digital music players, digital TV, and so on, as points of access to this vast ecosystem of networked content, communication, and control.

20.1.3 From Interaction to Experience

In a networked universe that has no center, we will use the human user as a subjective viewer of the whole experience ecosystem.

Around this hypothetical user, we will describe different concentric “spheres” of interaction. The first “sphere,” closest to the user, corresponds to what usually is taken in consideration when designing a computer system: the physical interaction with a device. The other “spheres,” growing progressively to include multiple elements of this experience ecosystem, will include topics related to human-to-human communication, or access to content in the “cloud,” or interacting with objects in a space.

When mobile devices are enriched by multiple functions and applications, this entire ecosystem is what actually influences the overall experience of the user. This is the new territory for designers to act and define the character of a product, the features of a service, or the identity of a medium.

20.2 SPHERES OF INTERACTION

20.2.1 Sphere 1: Of Humans and Devices

The first sphere of interaction, the most intimate connection between a human and a device, is the physical manipulation.

The very nature of handheld devices makes this manipulation a key component of the relationship between user and system. Although handheld devices like mobile phones, whether they have conventional keypads or a touch screen, sometimes mimic the navigation in the graphical user interface (GUI) of a computer, the overall experience is extremely different from using a PC. The manipulation and relation with the hand, and in some respect with the rest of the body (such as through placing the device in a bag or in the pocket), roots these objects in the history of tools that are real extensions of the hand, such as pens. Also, the handheld nature of mobile devices highlights social aspects, such as the proxemic role of the object, like handing over a phone to show a picture to a friend, for example. In this context, rich and sophisticated sociocultural aspects complement the basic ergonomic components of the interaction.

20.2.1.1 Cars and Motorcycles

If you talk to a passionate biker, you will most often hear a comparison between the emotional aspects of driving a motorcycle compared with the "dullness" of driving a car: riding a bike is a richer sensorial experience that involves the whole body. Although one drives a car by "codified" controls, proper interfaces that act on the machine, riding a bike involves an *extension* of the body.

Actually, riding a bike is a hybrid between some coded controls (such as clutch and brakes) and "natural" proprioceptive controls (such as moving the body while making a turn).

Consequently, a car may be perceived as passive, whereas riding a bike is perceived as an active and engaging experience. Furthermore, riding a bike puts you in direct, immediate connection with the surrounding environment. From a certain point of view, the same can be said of personal computers and cell phones or tablets. Computers' only relationship with the body is the flat space of the desk, the movements of fingers on the keyboard, and the hand on the mouse. We use computers while sitting alone at a desk, our bodies still, and our minds isolated from the surrounding environment. Mobile devices such as cell phones are handheld, and the relationship with the hands is a fundamental character of the interaction. A PC provides an experience that atrophies our body to minuscule movements of the fingers and the wrist, whereas handling a cell phone is an experience that may involve larger gestures and has a richer relationship with the body. We use mobile phones while walking, sitting, standing, and driving. We use them in isolation or in a crowd. The experience of communicating via a mobile phone is only partially immersive: the environmental noise and the distractions of the environment balance this immersion and provide meaningful context to our activity. Cell phones have transformed the way we approach others and the power of the context enriches the communication: rather than asking, "How are you?" when we start a conversation, we ask, "Where are you?" as if the environment was an integral part of the conversation. The limited size of the device, which is in some respects a limit to the interaction because the screen is small and crowded, actually helps provide a further element of context: with such a small screen, we are forced to see what is around it, and the interaction is always peripheral, never immersive, and always hybridized by the "real" experience of the world around.

20.2.1.2 Ritual Power of Hands

Consider the way people take pictures with a cell phone. Sometimes the camera phone is pointed toward the subject of the photo like a camera, and sometimes the phone is used in a very different way to take pictures that are much more spontaneous and transient. To be fair, the way camera phones are used had been anticipated by a phenomenon grown around very conventional film-based cameras. In the early 1990s, users of a camera called Lomo started to take very spontaneous pictures by not aiming in the viewfinder. The Lomo phenomenon transformed the genre of photography with a disruption that was purely "behavioral": a new gesture, such as handling the camera without bringing it near the eye, revolutionized photography, and made the images much more "instantaneous" and genuine. This gave birth to a new photographic genre that was called Lomography. In Lomography, the gesture *is* the photo. The way people manipulate camera phones is very similar. In this case, the instantaneous photo genre is reinforced by the fact that photos are taken to be shared, not to be kept. Although, in origin, photography was conceived as a way of "conserving" reality by documenting it, networked photos are transient, not perennial. And the informality of the gesture of "catching" them is a fundamental part of both the interaction and the narratives of this genre of photography. Designing a form that facilitated or denied this spontaneity would be for a designer a way of confirming, or denying, this behavior and, as a consequence, a way of influencing the genre of photographs taken with such a device. In this case, the whole form and shape of the device, rather than an interface or a control, is the fundamental component of the interaction. Better, the shape of the device *is* the interface. Although this is relatively new in the history of HCI, it is not new in the history of objects.

20.2.1.3 Form Follows Gesture

In the history of material culture, a history that includes, for example, ancient cooking utensils and working tools, the relationship between objects and hands is fundamental and ends up defining the ritual relationship between humans and their tools. We are facing a kind of "calligraphic" use of an object. The idea of calligraphy, etymologically a "beautiful way of drawing or writing," is very much related to the idea of dignity and richness of gesture. Touch-screen devices with multitouch and swipe leverage this idea by adopting gestures that go beyond the mere tap-to-select: they imply much more articulated gestures, and movements that discern pressure, acceleration, and chords, and react with the same rich articulation by providing visual dynamic feedback that include physics-like bounce, float, mass, inertia, and so on.

In addition to being a fundamental component of the use of the device, the overall shape, proportions, and mechanics of the object are also establishing the "character" of the object, and with that they influence the social dynamics around the usage of the object. The way John Travolta pulls the antenna of his flip phone with his teeth in Pulp Fiction (in the topical moment when his friend just shot a guy by error in a car) is integral part of his character, and of the violence in that scene. How different is this from the way Anna Magnani "hugs" the handset in a cinema rendition of Jean Cocteau's theatrical piece "La Voix Humaine," a love monologue "designed" around the relationship between the woman—a desperate lover who has been abandoned by her partner—and the handset of her phone, the only thread left between her and her lover.

The manipulation of handheld devices also has a social role: sharing a photo sometimes means also grabbing the digital camera or the phone and handing it to somebody to watch at the screen. Even if the screen is small, sharing it with a friend is a rich social act. Handheld devices are part of the proxemic relationship between humans, other humans, and the social space that surrounds them.

When devices such as cell phones are becoming so miniaturized that their shape is no more influenced by their technical components, their form would rather be defined by the way people manipulate them. And, as we described previously, the manipulation is only in part delimited by the mere rules of usability. With the multiplication of networked devices of different form and size, for example, different devices define their identity not just through their usability per se, but through their ability to find the right place in the context of use: touch-screen-based tablets like the iPad mimic the size of a magazine and they cover the interaction space between the "one-foot" interaction of a handheld and the "10-foot" interaction of TV.

20.2.1.4 Sense of Touch

The diffusion of keyboards and keypads has led to a kind of "buttonification" of the human being. This is definitely a reduction of the power of the sense of touch. Our senses are way more sophisticated than the basic on/off controls that they operate. In the case of handheld devices, this sensorial atrophy is more evident for at least two reasons. The first one is that, while we hold the device with the full hand, five fingers plus the palm, we only use one finger to actually operate the keypad and receive tactile feedback. The second one is that, for the "peripheral" nature of the interaction with handhelds that we mentioned before, our interaction with a keypad has in the past ignored other fundamental spatial and proprioceptual clues, such as the horizon, our wrist position, and gravity. The recent advances in sensing and feedback technologies have finally fought back this kind of sensorial deprivation that keypads have provoked. The introduction of accelerometers and gravity sensors allows the device to be "conscious" of its absolute position in space and its relative position to the hand of the user. This allows users to take full advantage of the proprioceptual character of handheld manipulation and provides a fuller tactile stimulation as a feedback to this manipulation. Similarly, with the introduction of haptics, the primitive vibrators used originally just to replace the ringer tone of phones are used to provide a broader tactile feedback. Haptic actuators amplified by the resonance of the whole mass of the handheld can stimulate the whole palm of the hand or they could be used in a denser "resolution" to provide localized feedback and a more meaningful haptic "message" to parts of the hand. When combined with a proper reflection of the proprioceptual character of the GUI, the overall combination of enriched tactile and visual contextualized experience finally takes full advantage of the handheld nature of mobile devices. Localized buttons seem to have disappeared completely from touch-based devices, and combining touch with these additional proprioceptual sensory messages will provide a much richer sensorial experience.

20.2.1.5 Product Species

Even when the interaction with devices is properly designed and based on the principles of usability, sometimes there seems to be an additional difficulty in understanding the functions of an object, a subtler barrier to its adoption. An object may fail to deliver a clear message of its functionality to the user. The issue in this case is not "How easy is it to perform a task with this device?" but "What does this device do?" or, even more, "What is this?". In the past, a good rule to introduce a new object has been to leverage an analogy with a known, familiar object.

The reference to familiar devices and interactions could still apply today, but can also become too big of a limitation, a restriction of new functionalities or richer paradigms that simply do not have any parallel with anything familiar. Too many references are challenged: networked TV has no channels; mobile phones and house phones hybridize; e-mail and instant messaging integrate; camera, telephone, and video functionalities converge in the same device. Communication paradigms such as social networking and blogging, or content distribution models such as podcasting, simply do not have any reference to past, established, familiar models.

Inventing new object archetypes, as much as it is an oxymoron, may be the only way out of this impasse—developing new product types and new interaction paradigms when playing analogy to existing archetypes is just not applicable. The same applies to monofunctional devices in comparison with multifunctional ones; the history of interaction design is full of successes of simple, monofunctional devices that leveraged this analogy.

From the Palm PDA to the iPod, all fall in the category of devices that "do one thing and do it right." In the future, however, reducing the functionality to achieve simplicity of use may be no longer possible. Multifunctionality may offer too much of an advantage to be compromised, and new generations of users, younger hyper-taskers, would not necessarily appreciate oversimplification.

So, the challenge for a designer is to ensure that an object would be "understood" while preserving multifunctionality and richness of experience. Designers would, on the one side, develop a richer perspective on visual affordances, one that could accommodate multiple and partially ambiguous affordances related to different usages. On the other side, beyond affordances, designers need to develop a better understanding of what constitutes a recognizable, understandable "species" of objects. That is to say, rather than seeking the familiarity with a known object or function "as-is," developing a more sophisticated understanding of how familiarity(ies), or just components of this, can be carried over along the evolution of an object, or carried across a discontinuity in the evolution of such an object or functionality. This tactic seems to have been adopted by the designers of the iPad, which uses the familiarity of the book, or newspaper, as a Trojan horse to enter in our houses with the purpose, in future, to introduce new behaviors and spurt the growth of a completely new species of devices, which may put the familiar book into oblivion. The result will be the transformation of the simple concept of affordance into a richer notion of "objecthood," defined as the ability of humans to recognize and perceive correctly the world of objects, and the mutual ability of objects to be understandable and to overtly disclose their functionality. That is, for both humans and objects to develop their relationship around a sixth sense, the "sense of meaning."

20.2.2 Sphere 2: Mental Models (and Their Representation)

20.2.2.1 Mental Models

In personal computers, the metaphor of the "desktop" and the "point and click" in that digital space starts to feel the "competition" of other models of interaction, namely the information architectures organized around e-mail and social networking, which today are often dominant compared with the desktop access to files and folders.

The mental construct of the interaction with mobile wireless media is intrinsically centered on communication; the phonebook, and its extension to include presence, status, and location, is often the master mental model that represents our communication universe. The phonebook acts as a point of departure for a broad series of actions: starting a call, generating a message, initiating a textual chat, posting our status or checking the ones of our "buddies," or sharing a picture. However, with the diffusion of other ways of contextualizing the interaction, such as location detection, a user may decide to switch to a different model and, for example, "view by location" and reorganize the communication, mapping it over the actual physical surroundings. In this model, a user may reorganize the phonebook to check first "who's around me" in that moment. Even if the idea of communicating in physical proximity may seem paradoxical, cell phones are often used to create an additional communication channel (e.g., texting or sharing images) with people who are not remote or to manage meetings with people in physical proximity (e.g., a group of friends in the same city arranging a dinner together).

Other mental models use incoming events as an organizational context: an incoming call may trigger a reorganization of the interface around the person who calls (bringing in the foreground previous messages or pictures exchanged with that person); or walking in front of a place may trigger some location-based content and contextual actions around that content; or an approaching event, such as a meeting, can trigger other information such as the names of the participants, their status, and their location.

Although the taxonomies arranged around the sphere of communication may work better for phone-first devices, the proliferation of functions (imaging, video, music management, e-mail, etc.) could suggest taxonomies based around a clear distinction of applications or around categories of content (my photos, my music, my phonebook, etc.).

The dominance of the former model (centered on communication) or the latter (centered on content or applications) is not yet clear. For a long time, we will probably continue to have hybrids with the copresence of multiple categorizations.

What is common to both these models and different from the old desktop metaphor is that they have a less direct analogy with an interaction space—such as the physical space of the desktop—and they have more reference to mental constructs. As such, these models could be referred to as "cosmographies." Although a metaphor like the desktop implies having a reference with something that exists, a cosmography is a "general description of the world or universe," and this could embrace descriptions of both the material world and of abstract concepts—a mental model of something that eventually does not have any relation with the material reality.

20.2.2.2 Greater Than Infinity

Aggregation seems to be one of the best techniques to manage large quantities of contacts or content. Many recent interfaces and services, for example, help users manage the exponential growth of social networking messages by aggregating them around a revisited version of the contact book. The "view by person" allows in this case accessing all the social updates and managing all the multiple identities of an individual. This confirms the validity of a mental model based on communication and content rather than a task-oriented approach that would split access to e-mails, phone calls, and social updates from an individual within different applications. Aggregation, however, falls victim of quantity: when all friends, and friends of friends, using different communication formats (e-mails, phone calls, social updates, content sharing, etc.) are aggregated into one single view, their quantity may become unmanageable. The aggregated contact book of a last-generation smartphone can easily exceed thousands of contacts, and the benefit of having them "at hand" in one place is hindered by the difficulty of managing such an almost-infinite list. For this reason, it is fundamental to develop techniques to manage quantities of data, whether contacts in a phonebook or videos in a channel lineup, that risk becoming a cognitive burden. Innovative mental models are among such techniques. In the example of an ever-growing phonebook, finding an alternative to the alphabetic A-to-Z structure is fundamental. This can be achieved by adopting innovative information structures (i.e., closer friends first, then friends of friends, in a concentrical, progressive logical geometry that would replace the linear, nonhierarchical

A-to-Z structure), or visualization techniques (i.e., visualizing the above mentioned structure in a center-periphery zoomable diagram), or relying on context to filter information (i.e., showing contacts relevant to the moment of the day or an upcoming calendar event).

20.2.3 Sphere 3: Context

20.2.3.1 Physical Context

Mobile interaction cannot be isolated from its context. Although the interaction between a user and a personal computer tends to be immersive and central and is agnostic of the physical context of the user, the interaction with mobile devices is typically contextual and peripheral.

Primitive graphic menus and icons arranged in the small screen, a kind of translation of the conventional PC, are still the dominant GUI paradigm in wireless mobile media, but they are only the most superficial aspect of the interaction with devices like cell phones. Multiple other aspects of interaction with cell phones are less tangible, yet more important, than the keypad-and-screen, or touch-based, user interface. The whole experience ecosystem of the interaction with wireless media includes the combination of the interaction with the physical device, with its services, and with the physical environment.

Context is what makes mobile interaction much richer than computer interaction. Mobility means to be *more connected* to the physical place, not *more disconnected* from it; communication relates to *where* we are and *who* is around us. Location-based services, mobile imaging, and mobile blogging extend this concept to content itself: content accessed and generated from a mobile platform has the potential to be heavily contextualized over the physical environment. The convergence of all these aspects leads toward the paradigm of augmented reality, where information is shown on the actual physical space. The full implementation of such paradigm would be the ultimate celebration of the contextual power of mobile interaction.

20.2.3.2 Smart Objects and Active Ambient

Smart tags are very tiny, low-cost, wireless microcomputers. They only cost a few cents and they can be attached to anything—items of clothing, food cans, spare parts for cars, electric irons, and so on. The tags can be programmed to provide information about the items to which they are attached. They are able to identify a product and network this information and can also take note of their environment, recording changes in temperature, and so on. Their broad adoption is changing the way logistics are organized around raw materials and products traveling to transformation and to markets.

Smart tags also enable a minute level of interaction with the physical space. Cell phones equipped with RFID tags are already enabling financial transactions such as paying a ticket on the underground or replacing a credit card. The extension of this level of interaction may further enrich the augmented reality paradigm by supporting the paradigm of "point and click at real environments." Handheld devices could be used to trigger contextual information when pointed at other objects or parts of the environment.

Finally, the connection of these tags to an integrated information system able to tell, at any moment, everything about the life of an item of produce, for example, may allow the quality, the origin, or the value to be tracked in real time. Produce that is fresh or older, in-season or off-season, near or far from its origin, with or without a track record of its growth will have a different value. In this scenario, smart tags may enable the creation of *digital word-of-mouth*, similar to the spreading of information and knowledge by human word-of-mouth, connected to the system of goods. Attached to items will be information and knowledge. It is possible to describe this as a *knowledge aura*, and this is the most interesting aspect of future material-knowledge systems. Experiments on connecting *collective wisdom* with digital systems on the field have been undertaken in India since the 1990s. The fully fledged scenario will allow replication in a digital environment of the wisdom of word-of-mouth know-how.

The whole idea of using advertisement before the purchase, and separate from the purchase experience, could be revolutionized by the presence of an aura of information that follows the product until the point of purchase. Through this aura of digital information, the product would "advertise itself" from the shelves of a shop. Even more intriguing is the idea that deeper information about a product, connected to similar experiences of purchase and use—what is known as a recommendation on a website—will be readily available at any moment a product is encountered. The lure and temptations that attract a buyer will come directly from the product, from the background voices of people who bought and used it, and by the life history of the product—a kind of very detailed, dynamic, digital certificate of authenticity.

20.2.4 Sphere 4: Communities

20.2.4.1 Interacting with Social Networks

Networked media are characterized by the hybridization of a multiplicity of communication genres. Although 20 years ago, the evolution of telephone technologies seemed to go toward hyperrealism—technologies focused on replicating, at best, face-to-face communication—the actual evolution of telecommunication has denied this trend and pointed toward mediating communication in a form *different* from face-to-face. Text messaging via SMS created a communication channel that was simply impossible to achieve with face-to-face—often more intimate, more enigmatic, than face-to-face, and complementary to that. Instant messaging has added the management of multiple one-to-one communications, also not possible in real face-to-face, and later, blogging created a hybrid between one-to-one expression and broadcast. Social networks put constant connection in a new light: they catalyze a permanent flow of status updates, a kind of collective *stream of consciousness*, that was unimaginable with old media.

Finally, the multimedia capabilities of devices like camera phones allow the integration between communication and content sharing, making it possible to share experiences in an almost permanent way.

Mobile wireless media are also the ideal platform to support tribal social structures: mobile phones have favored the birth of tribal forms of few-to-few and few-to-many communication, and they revitalize the scale of the small circle of friends, of groups of people who are in constant touch, exchanging information and passions, and sharing images and music.

Different from both an individual that communicates one-to-one with his interlocutor and from a broadcaster with a passive mass audience, tribal models of communication define a dense flow of exchanges within a restricted circle of friends. The space for tribal relations, superimposition of the physical space and of the "digital territory of belonging," also amplifies information and messages while building a shared memory. The permanence of content in a shared space creates not a simple exchange of messages but the sedimentation, the memory, and the construction of a shared permanent experience.

20.2.5 Sphere 5: Content

"Convergence" between media, for long just a buzzword, is happening for real, and this means a much richer interaction for the user. Richer communication—messaging, voice, and so on—is more and more integrated with content access, such as sharing music, videos, and images.

20.2.5.1 Deconstructed Content

Mobile imaging is an already-established form of hybridization between communication and content capture. Camera and video capture phones as an imaging platform are fully integrated with both the PC and the TV. Video and music content have already migrated to "the cloud," and this, combined with the integration of web content with TV productions, has created yet another "triple point" (mobile, PC, and TV and home systems) interaction with content.

The first consequence this will have from an interaction point of view will be the separation of content creation from content delivery and the consequent "deconstruction" of content. The possibility of capturing images with different devices (camera phones or digital cameras) and accessing them in different modalities (in the device itself, shared with others or published on the web, or printed or accessed from a PC or a TV) are already evident signs of separation between creation and access and examples of truly networked access to content. Mobile TV will not be an exact replica of TV delivered to home. The latter has already been transformed by Digital Video Recorders, and broadcasts have been separated from real time. Mobile TV will be even more disconnected from the real time of broadcast—often decomposed in smaller fragments, buffered, and stored locally or in the cloud—and reassociated with the "relative" time of the user; the possibility of caching content on a mobile device, combined with the need for "snacking" shorter versions of the same content during leftover time while commuting, for example, has made the idea of a personal, adapted palimpsest more realistic. Content is deconstructed in different multiple formats (filling up many lengths, between the short trailer and the full-length movie) and consumed from different platforms.

20.2.5.2 Narrowcast

Social networks, podcasting, and blogging are the signal of a larger trend that opens the possibility for everybody to become a publisher. In both cases, the broadcast paradigm evolves from one-to-many (where the one is an author, an institutional publisher such as a TV station) to any-to-many (where anybody, with a limited investment and a much leaner technology, can become a publisher). But there is more. Both blogging and podcasting are creating communities of adepts that do not behave like a conventional passive audience of broadcast mass media. They feed back, comment, re-tweet, and create a closer circle of communication that constitutes an actual "narrowcast" environment, a scale of publishing unheard of before. There is no solution of continuity between the one-to-one paradigm of telephony that extends to the larger scale of a tribe of friends and the blogging and podcasting narrowcast that takes broadcast to a smaller scale. Also, communication and content are so hybridized that they become indistinguishable.

20.3 SEAMLESS EXPERIENCE ECOSYSTEMS

Today, no interaction can be isolated in the confined space of HCI: any consideration on the interaction between a user and a mobile device cannot ignore the complexity of the relational "ecosystems" that exist around this interaction.

The components of this complex networked experience are as follows:

1. A social ecosystem, because mobile devices go beyond the one-to-one communication to mediate very rich and complex system of social relationships.
2. A content ecosystem, since mobile devices allow access to images, music, and videos, and peer-to-peer content generation and sharing introduce a completely new perspective of content authoring.
3. A spatial ecosystem, because location-based services and communication strongly "ground" the interaction via mobile devices in the physical space.
4. A business ecosystem, because no single company or designer can control the overall experience, but can only shape the interaction from a partial point of control.
5. A platform ecosystem, since mobile interaction is integrated with interaction with other platforms, such as personal computers and digital TV.

The complexity of these ecosystems puts additional challenges to designers who want to shape the interaction.

Among these challenges are the following:

Adapting experiences to evolving behaviors. Convergence will change drastically the way users perceive their sphere of content and communication and, sometimes, giving up familiar models to embrace new paradigms may actually facilitate the ease of use. This is in apparent contrast with one of the rules of user-centered design, which is to refer to familiar models and avoid disruptions. For example, the potential elimination of "channels" when TV content moves over IP may help embrace a new paradigm that has no reference to the old "zapping." In the same way, embracing the paradigm of one-phone-number-per-person may confuse users at first, but will actually facilitate the transition to a communication paradigm where the user, not the device, is the one to be reached. A well-crafted, user-oriented approach may distinguish between the respect of familiarities that are structurally embedded in human perception and pushing back resistance of old mental habits that may actually hinder innovation and cause disruptions.

Shaping an experience without controlling the full ecosystem. The design culture has long insisted on the protection of the brand and on establishing a "signature" experience to a device or a service. In the future, it will be very rare, however, that a single brand, thus a single designer, would control the full experience ecosystem. Two approaches to this are possible: one is a resistance to sharing components of the experience with other brands; another is embracing the complexity of the system and finding lean, smart ways to shape the experience without actually owning and controlling all the elements. The latter seems to be a more open-minded, relaxed way to accept changes in the environment a designer operates, whereas the former may end up in a stubborn, desperate attempt to protect a territory that simply cannot be protected anymore. In other words, it is better to adapt and shape an experience by seeding signature elements in few key areas rather than working on a more rigid model and risking losing all the control.

Embracing the idea of human-to-human (and human-to-content) mediation rather than HCI. The HCI discipline is rooted in the interaction between a human and a machine (or a system) and focused on the shift from a mechanical interaction to a logical one. Although this competence remains true for the broad majority of devices we are designing today, it is also true that another aspect, the one of designing interfaces to access other humans and content via a mediating interface, is becoming more and more dominant. A deeper understanding of human patterns of communication, and of the interactions between interfaces of devices and genres of communication, is needed. In the same way, the understanding of how innovative ways of interacting with content may shape new types and formats of content is also needed. Media studies were born in a condition in which the media genre (e.g., the TV programs), the transport technology (via aerial or cable TV), the device of access (the TV set), the method of interaction (zapping through channels), the method of delivery (broadcast), and the interaction environment (a couch potato in a living room) of a given medium were linked and homogeneous. Today, a certain type of content (e.g., a TV program) may be delivered in a multiplicity of formats (real time or cached, in snippets, or in a linear fashion) to extremely different platforms and environments (at home or while mobile, to a passive individual or annotated and redistributed within a community). This poses new challenges to designers that need to shape at the same moment an innovative interaction and an innovative medium genre.

Experience ecosystems are reshaping the way human knowledge is grown and diffused. Interaction designers have the responsibility of both shaping a pleasurable, meaningful experience and of facilitating the chain of innovations that flows through new behaviors, new business models, and new technologies. Designers need to be the first to drive innovation in their own disciplines.

21 Tangible User Interfaces

Hiroshi Ishii and Brygg Ullmer

CONTENTS

21.1 INTRODUCTION

Where the sea meets the land, life has blossomed into a myriad of unique forms in the turbulence of water, sand, and wind. At a different seashore, between the sea of bits and land of atoms, we are now facing the challenge of reconciling our dual citizenships in the physical and digital worlds. Our visual and auditory senses are steeped in the sea of digital information, but our bodies remain deeply rooted in the physical world. Windows to the digital world are confined within flat square screens and pixels, or "painted bits." Digital information remains caged apart from our visceral engagement, from enduring embodiments we can directly engage with our hands and bodies.

Imagine an iceberg, a floating mass of ice in the ocean. This is one metaphor of tangible user interfaces (TUIs). A TUI gives physical form to digital information and computation, salvaging the bits from the bottom of the water, setting them afloat, and making them directly manipulable with human hands.

21.2 FROM GRAPHICAL USER INTERFACE TO TANGIBLE USER INTERFACE

People have developed sophisticated skills for sensing and manipulating their physical environments. However, most of these skills are not used in interaction with the digital world today. A TUI builds upon these skills, situating physically embodied digital information within physical space. Much of the TUI design challenge surrounds extending the affordances of physical objects to express and support engagement with new associations, behaviors, and properties of the digital domain (Ishii and Ullmer 1997; Ullmer and Ishii 2000; Baskinger and Gross 2010).

Interactions with digital information are now largely confined to graphical user interfaces (GUIs). We are surrounded by ubiquitous GUI devices including personal computers, handheld computers, and smart phones. Graphical interfaces have been in existence in research form since the 1950s (Astrahan et al. 1957; Sutherland 1964; Thacker et al. 1979), first appearing commercially in the Xerox 8010 Star System (Smith 1982). With the commercial success of the Apple Macintosh and Microsoft Windows, the GUI has become the standard paradigm for human–computer interaction (HCI) today.

GUIs represent information (bits) with pixels on bitmapped displays. These graphical representations can be manipulated with generic remote controllers such as mice, touchpads, and keyboards. By decoupling representation (pixels) from control (input devices), GUIs provide the malleability to graphically mediate diverse digital information and operations. Utilizing these graphical representations and "see, point, and click" interaction, for many use contexts the GUI made significant usability improvements over its command user interface predecessor, which required the user to "remember and type" characters.

However, interaction with pixels on GUI screens is inconsistent with our interactions with the rest of the physical environment we inhabit. The GUI, bound to the screen, windows, mouse, and keyboard, is divorced from the ways interaction takes place in most physical contexts. Interacting with GUIs, we cannot take advantage of our dexterity and skills for physical manipulation, such as engagement with building blocks and shaping models out of clay.

TUIs aim to take advantage of these haptic interaction skills. The key idea of tangible interfaces is *giving physical form to digital information*. These physical forms serve both as representations and controls for their interwoven digital bindings and associations. TUIs make digital information directly manipulable with our hands, and perceptible through our foreground and peripheral senses.

TUIs serve as an alternative, and sometimes a complement to the current GUI paradigm, demonstrating a new path to materialize Mark Weiser's vision of Ubiquitous Computing (Weiser 1991). Weiser wrote eloquently of weaving digital technology into the fabric of physical environments, making computation invisible. Instead of melting pixels into assorted large and small screens of different devices, tangible interfaces seek an amalgam of thoughtfully designed visible materiality and cognitive transparency through diverse forms—soft and hard, robust and fragile, wearable and architectural, transient and enduring—situated throughout our varied physical habitats.

This chapter introduces the basic concept of TUIs, dimensions of tangibility, illustrative tangible genres and instances, and open design challenges and opportunities.

21.3 URP: A TANGIBLE EXAMPLE

As a first illustration of basic tangible concepts, we introduce Urp—the urban planning workbench (Underkoffler and Ishii 1999). Urp uses physical scale models of architectural buildings to configure and control an underlying urban simulation of shadow, light reflection, wind flow, and other properties (Photo 21.1). In addition to an ensemble of building models, Urp also provides a variety of interactive tools for querying and controlling parameters of the urban simulation. These tools include a clock to change the position of the sun, a material wand to change building surfaces between brick and glass (thus reflecting light), a wind tool to change wind direction, and an anemometer to measure wind speed.

Urp's physical building models cast digital shadows onto the workbench surface (using video projection). These correspond to solar shadows at a particular time of day. The time of day, entangled with the sun's position in the sky, can be controlled by turning the physical hands of a clock tool

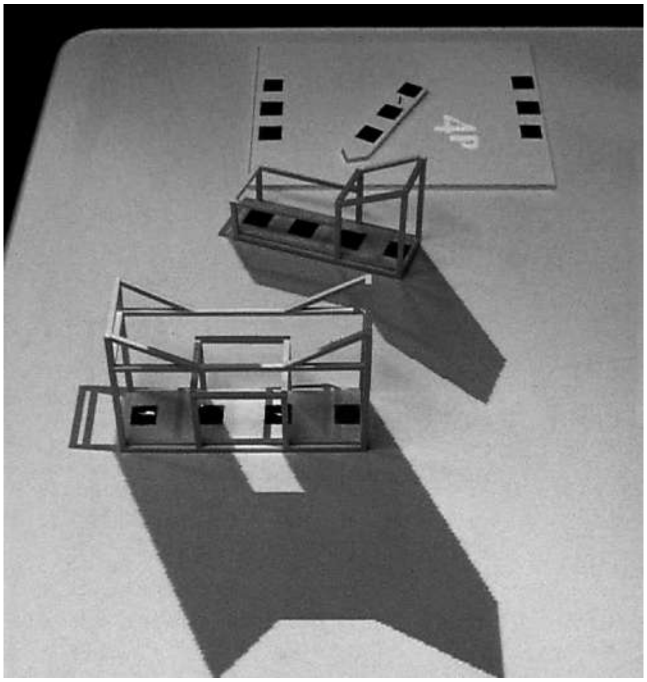


PHOTO 21.1 Urp and shadow simulation. Physical building models casting digital shadows, and clock tool to control time of day (position of the sun).

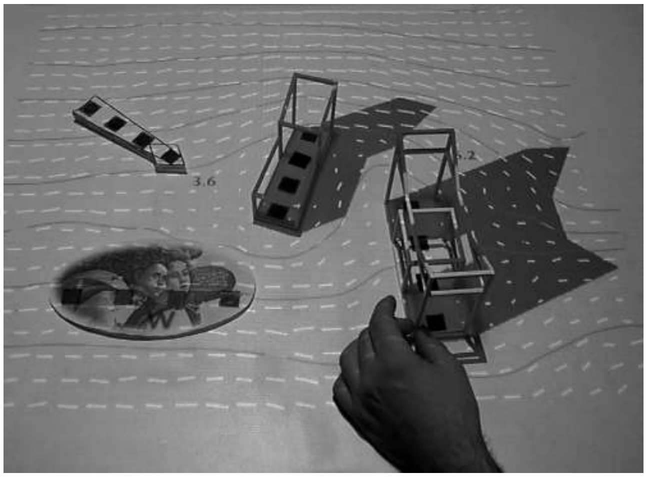


PHOTO 21.2 Urp and wind simulation. Wind flow simulation with wind tool and anemometer.

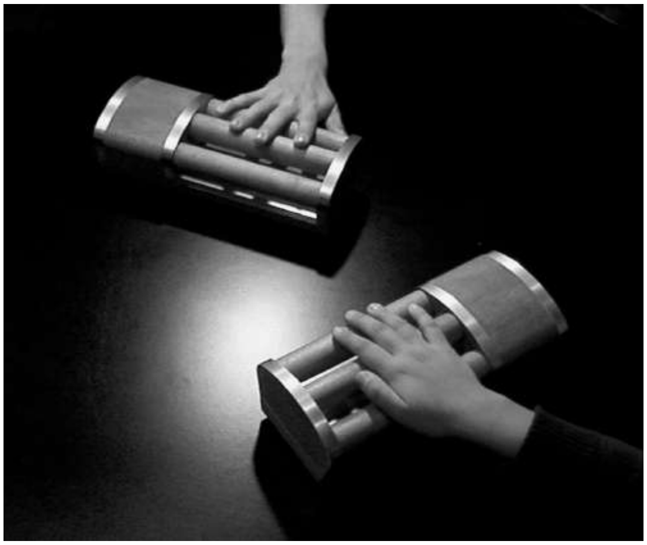


PHOTO 21.3 inTouch. inTouch is a prototype that explores tangible telepresence using touch over distance. Key idea is synchronizing distributed physical objects using force feedback technology.

(Photo 21.2). The building models can be repositioned and reoriented, with their graphically mediated solar shadows transforming according to their spatial and temporal configuration. Urban planners can identify and isolate intershadowing problems (shadows cast on adjacent buildings), and shape the play of urban light.

A material wand alters the surface properties of buildings. By touching the material wand to a building model, the surface material is transformed from brick to glass, and a simulated reflection of sunlight rebounds from the walls onto the worksurface (again through computationally mediated graphical projection). Moving the building allows urban designers to be aware of the relationship between building reflections and other infrastructure. For example, the reflection off the building at sundown might distract drivers on a nearby highway. The designer can experiment with altering the building's angles toward oncoming traffic, or moving the building further from the roadway. Tapping again with the material wand returns the material to brick; the reflected sunlight disappears, leaving only the projective solar shadow.

By placing the wind tool on the workbench, a windflow simulation is activated. This is based upon a real-time computational fluid dynamics simulation, with field lines graphically flowing around the buildings. Changing the wind tool's physical orientation correspondingly alters the orientation of the computational wind. Urban planners can identify potential wind problems, such as areas of high pressure that may result in challenging walking environments or hard-to-open doors (a real problem that partially disabled a building near our laboratory, until an iconic sculpture was commissioned to mold the wind). An anemometer tool allows monitoring of the wind speed (Photo 21.3). On placing the anemometer upon the workspace, the wind speed at that point is shown. Every few seconds, the corresponding numeric sails off along the flow lines, as with leafs upon the wind, to better visualize and quantify the evolving windflow. The interaction between the buildings and their environment allows urban planners to visualize and discuss intershadowing, wind, and placement problems.

In Urp, physical models of buildings are used as tangible representations of digital models of the buildings. To change the location and orientation of buildings, users grasp and manipulate the mediated physical model instead of pointing and dragging graphical representations on a screen with a mouse. The physical forms of Urp's buildings, and their configuration upon the workbench, represent and control the computational state of the urban simulation.

Although standard GUI devices such as keyboards, mice, and screens are also physical in form, the role of physical representation in TUI provides a key distinction. Urp's building models and tools are neither input nor output devices. Rather, these *tangibles* are manipulable representational physical embodiments of the simulation's key objects of interest. The building and tool tangibles are physical representations of both digital information (shadow dimensions and wind speed) and computational operations (windflow and shadowfall). Tangibles also serve as controls of the underlying computational simulation. Each specific physical embodiment supports a dual use in representing the digital model and allowing control of the digital representation.

In Section 21.4, a model of tangible interfaces is introduced, juxtaposed with a leading model for GUIs, to illustrate this mechanism.

21.4 BASIC MODEL OF TANGIBLE INTERFACES

Interfaces between people and digital information generally require two key components: input and output, or control and representation. *Controls* enable users to manipulate information, while *external representations* are perceived with the human senses. Figure 21.1 illustrates a simple such model of user interfaces consisting of control, representation, and information.

In the Smalltalk-80 programming language (Goldberg 1984; Burbeck 1992), the relationship between these components was proposed to follow a "model-view-controller" or "MVC" archetype. This has become a basic interaction model for GUIs.

Drawing from the MVC approach, we have developed an interaction model for both GUI and TUI. We carry over the "control" element from MVC, while dividing the "view" element into two subcomponents: tangible and intangible representations. We rename "model" to "digital information," toward generalizing this framework to illustrate differences between GUIs and TUIs.

In computer science, the term "representation" often relates to the programs and data structures serving as the computer's *internal* representation (or model) of information. In this article, the meaning of "representation" centers upon *external* representations—the external manifestations of information in fashions directly perceivable by human senses that include our visual, auditory, and tactile senses.

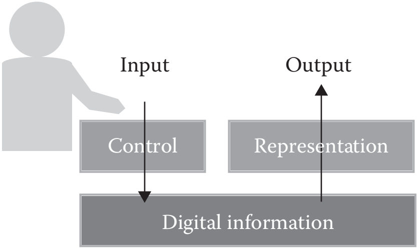


FIGURE 21.1 User interface. Interfaces between people and digital information requires two key components: 1) external *representations* (or view) that users can perceive, and 2) *controls* with which users can manipulate the representation.

21.4.1 Graphical User Interfaces

In 1981, the Xerox Star workstation set the stage for the first generation of GUIs (Smith 1982; Johnson et al. 1989), establishing the "desktop metaphor." This was initially modeled upon a partial simulation of an abstracted physical desktop, viewed upon a bitmapped screen. The Star workstation was the first commercial system to demonstrate the power of a mouse, windows, icons, property sheets, and modeless interaction. The Star also set several important HCI design principles, such as "seeing and pointing versus remembering and typing," and "what you see is what you get (WYSIWYG)." The Apple Macintosh brought this new style of HCI to the public's attention in 1984, creating a new trend in the personal computer industry. Now, the GUI is widespread, largely through the pervasiveness of Microsoft Windows and smart phones.

GUIs use windows, icons, and menus of pixels on bitmapped displays to mediate digital information. These are intangible representations. GUI pixels are traditionally rendered interactive through general "remote controllers" such as mice, touchpads and keyboards. In the pursuit of generality, GUIs typically introduce a deep separation between the digital (intangible) representation provided by the bitmapped display, and the controls provided by the mouse and keyboard.

Figure 21.2 illustrates the mainstream GUI paradigm in which generic input devices allow users to remotely interact with digital information. Using the metaphor of a seashore that separates a sea of bits from the land of atoms, digital information is illustrated at the bottom of the water, and mouse and screen are above sea level in the physical domain. Users interact with the remote control(s), and ultimately experience an intangible external representation of digital information (display pixels and sound).

21.4.2 Tangible User Interfaces

Tangible interfaces follow a different path than GUIs, using tangible representations of information that also serve as mechanisms for directly controlling digital information. Following articles including (Cohen, Withgott, and Piernot

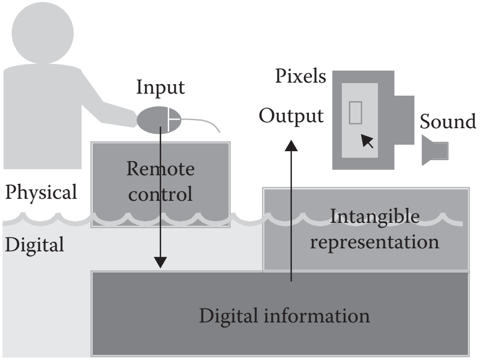


FIGURE 21.2 Graphical user interface. GUI represents information with intangible pixels on bitmapped displays and sound. General-purpose input devices allow users to control these representations.

1999; Singer et al. 1999; Ullmer and Ishii 2000), we will often refer to tangible representations as "tangibles." By representing information in both tangible and intangible forms, users can more directly control the underlying digital representation using their hands and bodies.

21.5 TANGIBLE REPRESENTATION AS CONTROL

Figure 21.3 illustrates the idea of tangible representation as control. Tangibles help bridge the boundary between the physical and digital worlds. Tangibles are computationally coupled to the control of the underlying digital information and computational models. Urp illustrates many examples of such couplings, including the binding of graphical geometries (digital data) to physical building models, and computational simulations (operations) to the physical wind tool. Instead of using a mouse to change the location and angle for graphical representations of a building model by pointing, selecting handles, and keying in control parameters, Urp users can grasp and move the building model to change both location and angle.

21.6 INTANGIBLE REPRESENTATION

While tangibles are coupled to digital information, they generally have limited direct ability to physically represent changes in their associated digital state. In comparison with malleable "bits," "atoms" are relatively inflexible. Unlike pixels on screens, today it remains comparatively difficult to change the form, position, or a property (e.g., color, size) of physical objects in real-time. (This may likely change in coming decades. Indeed, we will introduce several examples which already transcend these historical limitations.)

To complement this limitation of "atoms," TUIs also utilize malleable representations such as video projections and sounds, accompanying tangibles in their same physical space to give dynamic expressions of underlying digital information and computation. In Urp, the graphically mediated digital shadows and reflections that accompany physical building models are such an example.

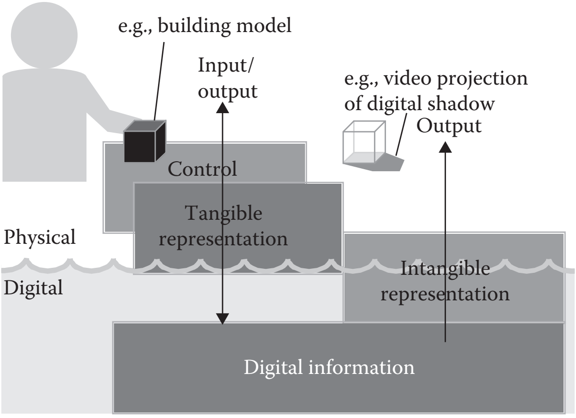


FIGURE 21.3 Tangible user interface. By giving tangible (physical) representation to digital information, TUI makes information directly graspable and manipulable with haptic feedback. Synchronous intangible representations (e.g. video projection) may complement tangible representations.

The success of a TUI often relies on a balance and strong perceptual coupling between tangible and intangible representations. In many successful TUIs, it is critical that both tangible and intangible representations are perceptually coupled to achieve a seamless interface that actively mediates interaction with the underlying digital information, and appropriately blurs the boundary between physical and digital. Coincidence of input and output spaces and real-time response are important elements toward accomplishing this goal.

Note: Some TUIs use actuation of tangibles as the central mean of computational feedback. Examples include inTouch (Brave, Ishii, and Dahley 1998), curlybot (Frei et al. 2000), and topobo (Raffle, Parkes, and Ishii 2004). This type of actuated TUI does not depend on "intangible" representation, as active feedback through the tangible representation serves as the main display channel.

21.7 KEY PROPERTIES OF TANGIBLE INTERFACES

While Figure 21.2 illustrates GUIs' clear distinction between graphical representation and remote controls, Figure 21.3 highlights TUIs' integration of physical representation and control. This model provides a tool for examining the following important properties and design requirements of tangible interfaces (Ullmer and Ishii 2000).

21.7.1 Computational Coupling of Tangibles to Digital Information

The central characteristic of tangible interfaces is the coupling of tangibles to underlying digital information and computational models. As corollaries, associated design challenges include (1) how to select or design legible, compelling tangibles that well-evoke their digital associations; and (2) how to best map tangibles and their manipulation to digital computation and feedback in a meaningful and comprehensible manner.

As illustrated by Urp, a range of digital couplings and interpretations are possible. Urp examples included coupling data to the building models, operations to the wind tool, and property modifiers to the material wand.

The embodiment and mapping of tangibles are influenced by application, audience, and the envisioned usage context. We will describe examples in which embodiments of varying specificity are used. In some applications, relatively abstract physical forms (such as round pucks) are used as generic controllers. These are typically reusable across varying bindings, allowing control over a variety of parameters through their rotation, depression, etc. (Patten et al. 2001). When a puck is used as a dial to control a simulation parameter, graphical feedback provides complementary information (e.g., the scale of the dial). In other cases, static bindings to physically representational tangibles can be a compelling alternative.

21.7.2 Tangible Embodiment of Mechanisms for Interactive Control

In addition to their representational aspect, tangibles serve as interactive physical controls. Tangibles may be physically inert, moving only as directly manipulated by a user's hands. Tangibles may also be physically actuated, whether through motors (e.g., inTouch, Curlybot), magnets (Pangaro et al. 2002), or other actuation methodologies (Biegelsen et al. 2000; Reznik and Canny 2001; Benhammous 2002).

Tangibles may be unconstrained and manipulated in free space with six degrees of freedom. They may also be weakly constrained through manipulation on a planar surface, or tightly constrained, as in the movement of abacus beads with one degree of freedom.

To make interaction simple and easy to learn, TUI designers need to utilize the physical constraints of the chosen physical embodiment. Because choices in physical embodiment interaction choices, designer should design interactions so that actions supported by tangibles are based on well-understood actions afforded by the artifact. For example, if a bottle-like form is chosen, opening the bottle by removing a cork or cap is a well-understood mechanism (Ishii, Mazalek, and Lee 2001). This understanding of culturally common manipulation techniques helps disambiguate the users' interpretation of how to interact with the object. Of course, these understandings vary widely between diverse cultures, presenting both opportunities and challenges.

21.7.3 Perceptual Coupling of Tangibles to Intangible Mediations

Tangible interfaces rely on a balance between tangible and intangible representations. Although embodied tangible elements play a central, defining role, TUIs' intangible representations play key supporting roles. A TUI's intangible representations—today, most often graphics and audio—often mediate much of the dynamic information provided by underlying computational processes.

Real-time feedback of intangible representations corresponding to tangible manipulations is critical toward supporting strong perceptual coupling. Coincidence of input and output spaces (spatial continuity of tangible and intangible representations) is also important to enhance perceptual couplings. For example, in Urp, the building tangibles are always accompanied by digital shadows (intangible representation) with minimal temporal or spatial gaps. This reinforces the illusion that the shadows are cast from the building models (rather than by the video projector).

21.8 DIMENSIONS OF TANGIBILITY

Building on the above properties, there are many dimensions surrounding the study and application of tangible interfaces. We next consider several of these, building upon frameworks and proposals by the broader research community. To be clear, these are not exhaustive, and is not as neatly classifiable as various categoricals amidst the hard sciences. We will then elaborate on several specific genres and examples of tangible interfaces in Section 21.9.

21.8.1 Conceptual

We have discussed tangible interfaces from the perspective of giving physical embodiments to digital information. A number of frameworks for tangible interfaces have been advanced. These have been recently considered at length in Mazalek and van den Hoven (2009) and Shaer and Hornecker (2009).

The term "tangible interaction" is a broader concept, of which tangible interfaces can be considered a particular approach. This perspective includes materiality, bodily interaction, and the embedding of systems in real spaces and contexts (Anderson et al. 2000; Djajadiningrat et al. 2002; Dourish 2002; Underkoffler and Ishii 1999; Hornecker and Buur 2006; Shaer and Hornecker 2009). Within this context, tangibility has often been approached from body (rather than information) centered perspectives (Dourish 2002; Klemmer, Hartmann, and Takayama 2006), with tangibles providing "resources for action" (Fernaeus, Tholander, and Jonsson 2008). Others have developed frameworks for tangibility surrounding the use of metaphors (Fishkin 2004; Hurtienne and Israel 2007), often drawing from embodiment and embodied action.

Another approach concerns the relationship between tangible interfaces and other physically situated forms of interaction (e.g., augmented, virtual, and mixed reality). One such perspective is reality-based interaction, which considers a fusion of physical-world properties and behaviors with bodily, environmental, and social awareness (Jacob et al. 2008). Tangible augmented reality is a specific example of crossover interaction techniques (Lee et al. 2004). In general, we see many opportunities for further understanding how tangible interfaces meet, engage, and hybridize with other styles of interaction.

21.8.2 Representational

As we earlier discussed, tangible interfaces are deeply concerned with the topic of representation. We have discussed earlier opportunities and tensions between tangible and intangible (e.g., graphical) representational forms. Representations in the physical world include a balance between shape and form (Djajadiningrat et al. 2004; Baskinger and Gross 2010), as well as static visual representations (Perlman 1976; Mackay and Pagani 1994; Perlman 1976; Ullmer et al. 2010). This could be seen as a tension and balance between body and skin. Tangibles can be general or specific in form and function and can use found/everyday/"readymade" objects (Goldsmith 1983; Mynatt 2000; Cheng et al. 2010) or purpose-designed forms (Frazer 1994; Gorbet et al. 1999; Djajadiningrat et al. 2004; Moggridge 2007; Baskinger and Gross 2010). Additional dimensions of representation (including related philosophical topics such as semiotics) are considered in Ullmer and Ishii (2000), Dourish (2002), Djajadiningrat et al. (2004), Hornecker and Buur (2006), and Shaer and Hornecker (2009).

21.8.3 Structural

Tangibles are influenced by a number of structural dimensions, influencing how they are shaped, constrained, and engage with each other and the world. The physical forms of tangibles may be rigid/solid, articulated, or malleable/plastic (e.g., as with clay, or more granularly as with sand). They may be used in the context of surfaces, edges, or spaces. Common surface examples include horizontal or vertical interactive surfaces, often illuminated graphically and sensed with computer vision or electromagnetic techniques (radio frequency identification). Wellner's DigitalDesk provided an influential early example and vision for interactive tabletops (Wellner 1993); we elaborate on this in Section 21.10.1. Ishii, Kobayashi, and Arita (1994) envisioned all the surfaces of architectural space—including walls, ceilings, windows, doors, and desktops—as active surfaces for engagement with remote partners and digital mediations.

Edges can serve to offer mechanical constraint; examples include linear racks and receptacles (Ullmer, Ishii, and Glas 1998; Cohen, Withgott, and Piernot 1999; Singer et al. 1999), rotary receptacles (MacLean, Snibbe, and Levin 2000; Ullmer, Ishii, and Jacob 2003), and material constraints such as strings and bands (Patten, Griffith, and Ishii 2000; Patten and Ishii 2007). In addition to their mechanical role as constraints, edges can partition architectural space, in the process establishing boundary regions—doorframes, hand rails, wainscoting, posts, columns, and culturally specific examples like the Japanese "genkan"—that can be awakened to computational mediation (Ullmer 2002).

There are also important structural implications in the ways that systems of tangibles are used together. In Ullmer and Ishii (2000), three such relationships were described: constructive, relational, and spatial (or positional). A fourth, associative, was also described, and has been developed further in van den Hoven and Eggen (2004), but is more semantic than structural in nature. We will discuss several of these in Section 21.9; they are also elaborated and extended in (Shaer et al. 2004; Shaer and Hornecker 2009).

21.8.4 Functional

Beyond the representational forms and structural relationships between tangibles lies the important question: what classes of functional roles can tangibles employ? Several generalizable conceptions of tangible functionality have been advanced. One approach is to consider physical instantiations of the GUI metaphor (Ishii and Ullmer 1997; Ullmer and Ishii 1997). Here, GUI windows, icons, menus, handles, and widgets can be physically instantiated as lens, phicon (physical icon), tray, handle, and instrument tangibles.

Another important approach advanced a model of containers, tokens, tools, and faucets (Holmquist, Redstr, and Ljungstrand 1999). Containers are tangibles that can be dynamically bound to instances or ensembles of digital information. The marbles of Bishop's well-known Marble Answering Machine (Crampton Smith 1995) that serve as containers for voice messages offer an early compelling example. As in other TUIs, Bishop used marble containers as a medium for moving information both within individual tangible products, and between systems of such products.

Holmquist et al. proposed "tokens" as a term for physically representational tangibles that are permanently bound to specific information bindings. The phicon term has also been used for this role (Ullmer and Ishii 1997), but without regard to whether the tangibles are abstract or representational in form, and dynamic or static in their physical–digital couplings. The "token" term has also been used more generically to refer to tangibles (Ullmer and Ishii 2000; Shaer et al. 2004); "totem" could be another alternative term (Lin 2007). Holmquist et al. consider tools as tangibles for binding to digital operations and faucets as access points for mediating the digital associations of tangibles (be they graphical, audible, haptic, olfactory, etc.).

Other researchers have discussed human language and parts of speech as inspirations for modeling the functionality of tangibles. Underkoffler and Ishii discuss nouns, verbs, attributes, and reconfigurable tools as tangible meanings within Urp (Underkoffler and Ishii 1999). Fishkin also describes noun and verb roles, both metaphorically and functionally (Fishkin 2004).

In another variation, Koleva et al. describe six different categories of tangibles: tools (specialized and general purpose), identifiers, proxies, projections, and illusions-of-same-object (Koleva et al. 2003). Edge and Blackwell make important contributions from the perspective of cognitive dimensions (Blackwell 2003; Edge and Blackwell 2006), grounding their analyses on tangible programming languages.

Development of functional models seems important for the future of tangible interfaces, both toward enabling rich behaviors within individual systems, and fostering interoperability between diverse systems (Ullmer et al. 2008; Ullmer et al. 2010). The above provide a number of promising points of departure; in parallel, we see this as an area where further progress is necessary.

21.8.5 Social

In the context of physical spaces and human relationships, it is known that distance matters (Olson and Olson 2000). Working in adjoining rooms, floors, buildings, or continents has quite different implications for human communications and collaborations (Kraut, Galegher, and Egido 1988). These observations have implications for entanglements between tangible interfaces and human social interactions. As cohabitants within physical space, tangible interfaces have a substantially different relationship with people than their screen-based kin. This relationship varies depending on physical locality, on whether proximal humans are alone or in groups, whether group interaction is collocated or distributed, and on cultural context (Hall 1969; Lawson 2001), among other social implications. Both as individuals and groups, people associate socially informed meanings with things (Csikszentmihaly and Rochberg-Halton

1981; Zigelbaum and Csikszentmihalyi 2009). In collocated groups, physical artifacts are important conversational mediators and facilitators (Hutchins 1995).

Building upon these and related observations, tangibles have been found to enable collocated collaborations that have been difficult to mediate with graphical interface alternatives (Streitz et al. 1999; Ben-Joseph et al. 2001; Eden, Scharff, and Hornecker 2002; Jacob et al. 2002; Hornecker and Buur 2006; Malone et al. 2009). Similar observations and opportunities have also been observed for tangibles mediating distributed collaboration (Brave, Ishii, and Dahley 1998) in both professional (Ishii, Kobayashi, and Arita 1994; Kuzuoka and Greenberg 1999; Moran et al. 1999; Singer et al. 1999; Everitt et al. 2003) and domestic contexts (Dunne and Raby 1994; Strong and Gaver 1996; Dodge 1997; Hindus et al. 2001; Chang et al. 2002). Tangible applications for distributed collaboration include intersections with genres of social media that have recently proved highly popular in mainstream culture (Kuzuoka and Greenberg 1999; Singer et al. 1999; Donath and Boyd 2004; Mackay and Beaudouin-Lafon 2005; Etter and Roecker 2007; Kalanithi and Michael Bove 2008).

21.9 GENRES OF TANGIBLE USER INTERFACE APPLICATIONS

TUIs have been used within a wide variety of application domains. This section gives an overview of seven genres for promising TUI applications. For a more exhaustive survey of TUIs in historical context, we encourage readers to refer to Ullmer and Ishii (2000) and Shaer and Hornecker (2009).

21.9.1 Constructive Assemblies

One of several structural approaches, the constructive assembly approach draws inspiration from LEGO and building blocks, centering upon the assemblage and interconnection of modular physical elements. This subdomain is typically concerned with the mechanical interconnection and kinetic relationships between systems of tangibles.

The constructive assembly approach was pioneered by Aish and Frazer in the late 1970s and early 1980s. Aish developed BBS for thermal performance analysis (Aish 1979; Aish and Noakes 1979). Frazer developed a series of intelligent modeling kits such as “Universal Constructor” (Frazer, Frazer, and Frazer 1980; Frazer 1994) for modeling and simulation. Additional examples include Geometry-Defining Processors (Anagnostou et al. 1989), AlgoBlock (Suzuki and Kato 1993), Triangles (Gorbet, Orth, and Ishii 1998), Blocks (Anderson et al. 2000), ActiveCube (Kitamura, Itoh, and Kishino 2001), and System Blocks (Zuckerman and Resnick 2004). Topobo (Raffle, Parkes, and Ishii 2004) inherited properties from both “constructive assemble” and “tangibles with kinetic memory” (Section 21.9.6).

21.9.2 Tokens and Constraints

“Tokens and constraints” is another structural approach, often concerned with tangibles linked to abstract digital information and manipulated using mechanical constraints (Ullmer, Ishii, and Jacob 2005). In this context, tokens are discrete, spatially reconfigurable tangibles that represent digital information or operations. Constraints are confining regions within which tokens can be placed. Constraints are mapped to digital operations or properties that are applied to tokens placed within their confines. Constraints are often embodied as physical structures that mechanically channel how tokens can be manipulated, often limiting their movement to a single physical dimension.

The Marble Answering Machine (Crampton Smith 1995) is a classic example that influenced many following research efforts. MediaBlocks (Ullmer, Ishii, and Glas 1998), LogJam (Cohen, Withgott, and Piernot 1999), DataTiles (Rekimoto, Ullmer, and Oba 2001), and tangible query interfaces (Ullmer, Ishii, and Jacob 2003) are other examples of this genre.

21.9.3 Interactive Surfaces

Interactive surfaces are likely the most popular structural approach for tangible interfaces. Both horizontal interactive surfaces, sometimes called tabletop TUIs or tangible workbenches, and vertical interactive surfaces/walls are common variations. Collaborative design, simulation, and performance, typically keying on the spatial/positional configurations of tangibles, have been explored by numerous researchers. In a typical augmented workbench, discrete tangible objects are manipulated, with their movements sensed by the workbench. Visual feedback is provided on the workbench surface, with coincident input/output space updated in real-time.

Digital Desk (Wellner 1993) is a pioneering work in this genre. Other examples include Bricks (Fitzmaurice, Ishii, and Buxton 1995b), metaDESK (Ullmer and Ishii 1997), InterSim (Arias, Eden, and Fischer 1997), Illuminating Light (Underkoffler and Ishii 1998), Urp (Underkoffler and Ishii 1999), Build-It (Rauterberg et al. 1998), Sensetable (Patten et al. 2001), AudioPad (Patten, Recht, and Ishii 2002), IP Network Design Workbench (IPNWDWB) (Kobayashi et al. 2003), Reactable (Jorda et al. 2007), and Lumino (Baudisch, Becker, and Rudeck 2010). Among wall-oriented systems, the ClearBoard was an early work (Ishii, Kobayashi, and Arita 1994), with LegoWall (Fitzmaurice 1996), Collaborage (Moran et al. 1999), Designer’s Outpost (Klemmer et al. 2001), and SenseBoard (Jacob et al. 2002) as additional well-known examples.

21.9.4 Continuous/Plastic Tangible User Interfaces

One limitation of early TUIs was an inability to change the forms of tangible representations during interaction. Users engaged predefined sets of fixed-form objects, changing the spatial relationship between them but not the forms of tangibles themselves. As an alternative, some TUI systems utilize continuous tangible materials such as clay and sand. Examples include Illuminating Clay (Piper, Ratti, and Ishii 2002), SandScape (Ishii et al. 2004), Phoxel-Space project (Ratti et al. 2004), and Digital Clay (Reed 2009).

21.9.5 Tangible Telepresence

Another tangibles genre concerns the mapping of haptic input to haptic representations over a distance. Also called "tangible telepresence," an underlying mechanism is the synchronization of distributed objects and the gestural simulation of "presence" artifacts, such as movement or vibration. These can allow remote participants to convey haptic manipulations of distributed physical objects. One outcome is giving remote users the sense of ghostly presence, as if an invisible person was manipulating a shared object. inTouch (Brave and Dahley 1997), HandJive (Fogg et al. 1998), and ComTouch (Chang et al. 2002) are illustrative examples.

21.9.6 Tangibles with Kinetic Memory

The use of kinesthetic gestures and movement is another promising application genre. For example, educational toys promoting constructionist learning concepts have been explored using actuation technology, taking advantage of tangible interface's input/output coincidence. Gestures in physical space can illuminate symmetric mathematical relationships in nature, and kinetic motions can be used to teach children concepts ranging from programming and differential geometry to storytelling. Curlybot (Frei et al. 2000) and topobo (Raffle, Parkes, and Ishii 2004) are examples of toys that distill ideas relating to gesture, form, dynamic movement, physics, and storytelling.

21.9.7 Augmented Everyday Objects

Augmentation of familiar everyday objects is an important approach for tangible interfaces to lower the design and conceptual floor, and make it easier to access and engage with basic tangible concepts. Early context from DuChamp's readymades is described in Goldsmith (1983) and Holmquist, Schmidt, and Ullmer (2004). Tangible interface examples include the Paper Audio Notebook (Stifelman 1996), musicBottles (Ishii et al. 1999), HandScape (Lee et al. 2000), Webstickers (Ljungstrand, Redström and Holmquist 2000), Everyday Media (Mynatt 2000), LumiTouch (Chang et al. 2001), Designers' Outpost (Klemmer et al. 2002), I/O Brush (Ryokai, Marti, and Ishii 2004), and iCon (Cheng et al. 2010). As evidenced by DuChamp (Goldsmith 1983), this genre seems particularly open to interpretation by artists and designers, toward questioning and engaging how everyday physical artifacts evolve with technology.

21.9.8 Ambient Media

In our early stages of tangibles research, we explored ways of improving the quality of interaction between people and digital information. We used two approaches for extending interaction techniques to the physical world:

- Allowing users to "grasp and manipulate" foreground information by coupling bits with physical objects
- Enabling users to be aware of background information at the periphery using ambient media in an augmented space

At that time (the mid-1990s), HCI research had focused primarily on foreground activity on the screen, neglecting the rest of the user's computing environment (Buxton 1995). However, people subconsciously receive ambient information from their peripheral senses without attending to these explicitly. If something unusual is noticed, it is awakened to our attention, and we can decide whether to bring it to the foreground. For example, people are subconsciously aware of the weather outside their window. If the user were to hear thunder or a sudden rush of wind, they can sense from their peripheral attention that a storm is on its way. If they wished, they could then look outside, attending to details, or continue working without distraction.

Ambient media describes a class of interfaces that is designed to smooth the transition of the user's focus of attention between background and foreground. For example, Natalie Jeremijenko's Live Wire (at Xerox PARC, 1995) was a spinning wire that moved to indicate network traffic. Designing such simple, legible representations for ambient media using tangible objects is a key part of the challenge of Tangible Bits (Ishii and Ullmer 1997).

The ambientROOM was a project that explored the ideas of ambient media, based upon a special room equipped with embedded sensors and ambient displays (Ishii et al. 1998). This work was a preliminary investigation into background/peripheral interfaces, and led to the design of standalone ambient fixtures such as Pinwheels and Walter Lamp that make users aware of "digital wind" and "bits of rain" at their peripheral senses (Dahley, Wisneski, and Ishii 1998).

Strictly speaking, ambient media lives at the edges of the TUI design space, as in many cases there are no direct interactions. Nonetheless, ambient media can serve as background information displays complementing tangible/graspable media that users manipulate in their foreground. The TUI approach to ambient media is concerned with design of simple mappings that give easy-to-understand form to information from cyberspace, representing related changes in subtle ways.

This concept of "ambient media" is now widely studied in the HCI community as a way to turn architectural/physical spaces into ambient, calm information environments (Wisneski et al. 1998; Redstrom, Skog, and Hallnas 2000; Holmquist and Skog 2003; Mankoff et al. 2003; Pousman and Stasko 2006). Another design space is low-attention interfaces for interpersonal communication through ambient media (Chang et al. 2001). This area has also been the subject of several commercialized products, such as Ambient Devices' Ambient Orb and Weather Beacon (http://www.ambientdevices.com/), and GlowCaps from Vitality (http://www.vitality.net/).

21.10 TANGIBLE USER INTERFACE INSTANCES

Next, several specific examples of tangible interfaces are presented both to illustrate the potential application domains described in Section 21.9 and to highlight unique features of TUIs. Since the mid-1970s, hundreds of tangible interfaces have been realized from scores of groups on every continent. Although not fully representative of broader community efforts, we describe a set primarily from the Tangible Media group's work of the past 15 years. We have chosen a subset that is illustrative of broader design spaces, as examples where our interpretations can go beyond the original published content. For a more broadly representative set of examples, we refer the reader to Ullmer and Ishii (2000), Mazalek and van den Hoven (2009), Shaer and Hornecker (2009), and Baskinger and Gross (2010). As illustrative examples beyond those described elsewhere in this chapter, we commend the reader to Reactable (Jorda et al. 2007; Jorda 2008), ISH (Hummels and Helm 2004), the Slot Machine (Perlman 1976; McNerney 2004), MONIAC (1952), Passive Props (Hinckley et al. 1994), Environment Audio (Oba 1990; Ullmer 2002), DataTiles (Rekimoto, Ullmer, and Oba 2001), Meatbook (Levisohn et al. 2007), CabBoots (Frey 2007), TinkerSheets (Zufferey et al. 2009), Lumen (Poupyrev, Nashida, and Okabe 2007), GeoTUI (Couture, Riviere, and Reuter 2008), Lilypad (Buechley et al. 2008), Siftables (Merrill, Kalanithi, and Maes 2007), BeatBearing (Bennett 2010), and SMSlingshot (Fischer et al. 2010).

21.10.1 DigitalDesk: A Pioneer of Tabletop Tangible Interfaces

DigitalDesk (Wellner 1993) was a pioneering system and vision that demonstrated ways to integrate physical and digital document processing on a table. Wellner brought functionality we typically associate with GUIs onto the physical desktop. He used a camera and a microphone to detect finger interaction on a graphical interface displayed on a desk with a video projector. Wellner used this desk for tasks such as graphic design and spreadsheet computations involving printed physical paper. This system also used and envisioned several kinds of physical props.

DigitalDesk illustrates compelling reasons for considering computer interfaces based on horizontal interactive surfaces. With many worksurfaces in our environment already existing as planar horizontal surfaces, integrating computation into these surfaces may provide opportunities for new types of relationships between cyberspace and physical objects. This juxtaposition may also help create computer systems that are more relevant to problem domains with established tabletop-based work practices. The DigitalDesk inspired many tabletop tangible interfaces including the Luminous Room (Underkoffler, Ullmer, and Ishii 1999), Urp (Underkoffler and Ishii 1999), and Sensetable (Patten et al. 2001).

21.10.2 Sensetable and AudioPad: A Tabletop TUI for Real-Time Music Performance

Sensetable (Patten et al. 2001) is a system that wirelessly tracks the positions of multiple objects on a flat display surface. Sensetable serves as a common platform for a variety of tabletop TUI applications such as AudioPad and the IPNWDWB.

AudioPad (Patten, Recht, and Ishii 2002) is a composition and performance instrument for electronic music. It tracks the positions of tangibles on a tabletop surface, and uses physical interactions with these to performatively engage with electronic music. AudioPad allows performers to pull sounds from a large repertoire of samples; juxtapose archived recordings against warm synthetic melodies; cut between drum loops to create new beats; and apply digital processing, all in parallel on the same physical table. AudioPad both allows for spontaneous reinterpretation of musical compositions, and creates a visual and tactile dialog between itself, the performer, and the audience.

AudioPad is based on the Sensetable platform. This uses a 2D array of antenna elements to track the positions of tagged tangibles. Each tangible represents either a musical track or a microphone (Photo 21.4). Software translates the identity and position of these tangibles into music and graphical feedback.

Experience with AudioPad in a series of live performances suggests that tabletop interaction with musically and graphically mediated tangibles can be a powerful and satisfying medium for musical expression. The seamless integration of input and output spaces allows the performer to focus on making music, rather than using the interface. The spatially multiplexed inputs of TUI also provide a compelling means for multiple musicians to collaboratively perform (Photo 21.5).

21.10.3 IP Network Design Workbench: An Event-Driven Simulation on Sensetable

The IPNWDWB is a collaborative project between NTT Comware and the Tangible Media Group. The IPNWDWB supports collaborative network design and simulation by groups of experts and customers (Kobayashi et al. 2003). This system is also based on the Sensetable platform, which wirelessly detects the location and orientation of physical pucks. The simulation engine is based on an event-driven simulation model. Using Sensetable, users can directly model and manipulate network topologies, control node and link simulation parameters, and simultaneously see simulation results projected onto the table in real-time (Photo 21.5).

The goal of IPNWDWB is to make simulation tools more accessible for nonexperts, so they can join the network design process and interact with experts more easily than when using traditional GUI computers. This system was commercialized and has been used for collaborative network design with customers to ensure their understanding of the performance and cost of network enhancements—for example, concerning

PHOTO 21.4 AudioPad running on Sensetable platform. Audiopad is a composition and performance environment for electronic music. Tangibles are tracked on an interactive tabletop system; their evolving physical/digital state is transformed into music.

PHOTO 21.5 IP Network Design Workbench running on Sensitable platform. The IP Network Design Workbench supports collaborative network design and simulation by multiple experts and customers on the Sensetable platform using an event-driven simulation engine.

increases in network traffic caused by Voice over IP and/or streaming video. The large horizontal work surface and tangible interaction invites participants to touch and manipulate pucks simultaneously, helping the decision-making process to become more democratic and convincing.

Comparing IPNWDWB with Urp, we see a substantial difference in the nature of applications. In Urp, the principle tangibles were physical scale models of buildings, which humans have used for 1000 years in the design of cities. It is natural to apply TUIs to such domains where physical models have been used and surrounding work practices established long before the birth of digital computers.

In contrast, IP Network Design is based on event-driven simulation models that are quite abstract, new, and largely dependent on computers. IPNWDB is important in its demonstration that tangible interfaces can empower the design process even for abstract application domains that do not have straightforward mappings from abstract concepts to physical objects. A wide range of modeling and simulation techniques, such as system dynamics and event-driven simulation, use 2D graph representation. We learned that many such applications can be supported by Sensetable-like TUI platforms in collaborative design sessions. For example, simultaneously changing parameters, transferring control between different people or different hands, and distributing the adjustment of simulations dynamically are interactions enabled by TUI.

21.10.4 Actuated Workbench: Closing the Loop between Sensing and Computational Actuation

The tabletop tangible interfaces we have discussed share a common weakness. While input occurs through the physical manipulation of tangibles, output is displayed only through sound or graphical projection on and around the objects. As a result, the tangibles can feel like loosely coupled handles to digital information, rather than physical manifestations of the information itself.

In addition, users must sometimes compensate for inconsistencies when links between digital data and tangibles are broken. Such broken links can arise when a change occurs in the computer model that is not reflected in a physical change of its associated object. With the computer system unable to move the tangibles, it cannot physically undo user input (e.g., restore the physical state of tangibles), correct physical inconsistencies in tangible layouts, or physically guide the user in the manipulation of tangibles. As long as this is so, the physical interaction between human and computer remains one-sided.

To address this problem, the Actuated Workbench was designed to provide a hardware and software infrastructure allowing tangibles to be moved on an interactive surface in two dimensions under computational control (Pangaro et al. 2002). The Actuated Workbench uses an array of magnets to move one or multiple objects upon a table. It is intended for use with existing tabletop tangible interfaces, providing an additional feedback loop for computer output and helping to resolve inconsistencies.

Actuation enables a variety of new functions and applications. For example, a search and retrieve function could respond to a user query by (say) moving matching items to another place on the tabletop, or wiggling them to seek the user's attention. Going a step further, a set of tangibles could, under computational control, be physically sorted and arranged on the table according to user-specified parameters. This could help the user organize a large number of data items, sets, or databases (each equally appropriate for tangible embodiment) before manually interacting with them. As a user makes changes to data through physical interaction, he or she may wish to undo some changes. A physical undo could move tangibles back to their positions before the last change. It could also show the user the exact sequence of movements he or she had performed. In this sense, both "undo" and "rewind" commands are possible.

One advantage of tabletop TUIs is the ease with which multiple users can make simultaneous changes to the system. Users can observe each other's changes, with any user reaching out and physically changing the shared layout. The situation is altered when users collaborate remotely. Here, a mechanism for physical actuation of tangibles becomes valuable for synchronizing tangibles upon physically separated workbenches (Photo 21.6). Without such a mechanism, real-time physical synchronization of the tables would not be possible, and inconsistencies could arise between the graphical projection and the tangible's physical state.

In addition to facilitating synchronization, the Actuated Workbench can recreate remote users' gestures with objects on the table, adding considerably to the "ghostly presence" sought in remote collaboration interfaces (Brave, Ishii, and Dahley 1998). This approach is in resonance with broader trends in embodied telepresence (Paulos and Canny 1998; Jouppi et al. 2004).

As an example application, Actuated Workbench was found to be helpful in teaching students about physics by demonstrating attraction and repulsion of charged particles (represented by pucks on the table). As students moved tangibles about the table, the system could make them pull together or push apart, illustrating the forces between the objects. A major next-generation system built upon the workbench platform, Pico, added both actuated and passive mechanical constraints, and realized a cell phone tower placement domain application (Patten and Ishii 2007) (see Photo 21.6).

21.10.5 SandScape: A Continuous Tangible User Interface for Landscape Design

SandScape (Ishii et al. 2004) is a tangible interface for designing and understanding landscapes through the combination of physical sand and a variety of computational simulations. Users view these simulations projected on the surface of a sand model representing terrain. Users can choose from

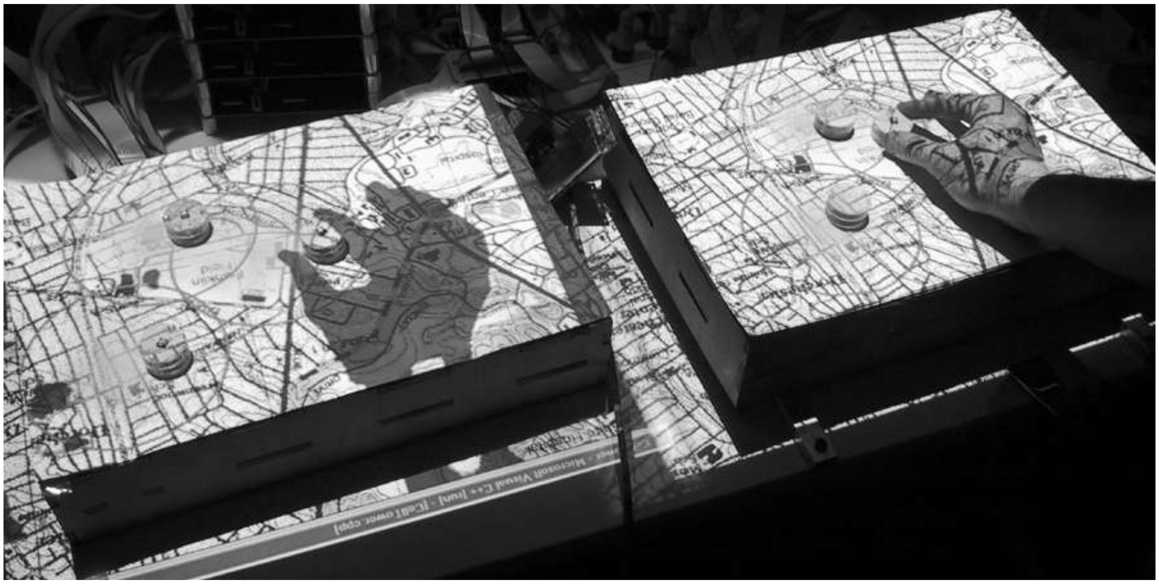


PHOTO 21.6 Actuated Workbench used for distributed collaboration. A mechanism for physical actuation of the pucks becomes critical for synchronizing multiple physically separated workbench stations.

a variety of different simulations that highlight the height, slope, contours, shadows, drainage, or aspect of the landscape model (Photo 21.7).

Users can alter the form of the landscape model by manipulating sand, seeing the resultant effects of computational analysis generated and projected on the sand's surface in real-time. The project illustrates how TUIs can leverage our ability to understand and manipulate physical forms, while still harnessing the power of computational simulation to help our understanding of model systems.

Our SandScape configuration was based on a box containing millimeter-diameter glass beads, lit from beneath with an array of 600 high-power infrared light-emitting diodes (LEDs). Four IR mirrors were placed around the LED array to generate a more even radiance distribution. A monochrome IR camera was mounted 2 meters above the surface of the beads, capturing the intensity of light passing through the volume. The intensity of transmitted light is a function of the depth of the beads; a look-up table was used to convert surface radiance values into surface elevation values. The system was calibrated to work with a specific bead size and type. SandScape is less accurate than its predecessor, Illuminating Clay, which used laser range finder to capture the geometry of a clay model; but at the time, the infrared approach was roughly one-hundredth the cost of the laser scanner (Piper, Ratti, and Ishii 2002).

SandScape and Illuminating Clay illustrate some of the potential advantages of combining physical and digital representations for landscape modeling and analysis. The physical clay and sand models convey spatial relationships that can be intuitively and directly manipulated by hand. Users can also insert any found physical objects directly under the camera. This approach allows users to quickly create and understand highly complex topographies that would be difficult and time-consuming to produce with conventional CAD tools. We believe this "continuous/plastic TUI" approach makes better use of our natural abilities to discover solutions through the manipulation of physical objects and materials.

In parallel, projected graphics give users real-time feedback. While tracked physical models interfaced with computers were not in themselves novel, we believe that SandScape and Illuminating Clay offered a new contribution by using the model's continuous surface geometry itself as the input/output mechanism. In this way, we hoped to give projective intangible information the same tangible immediacy as the clay/sand material itself, allowing quantitative data to support an intuitive understanding of the landscape.

Landscape architecture, as well as urban and architectural design, requires the collaboration of multiple specialists. These include earth engineers, water engineers, agrarian managers, land economists, transport engineers, and beyond. In the current design process, collaboration happens at different stages, sometimes without much direct interaction. SandScape and Illuminating Clay provide a common platform for collaboration centered on the table workspace. Numerous representations and analyses can be combined in a single design environment, potentially offering greater cohesion between different specialists and streamlining the design process.

21.10.6 MusicBottles: A Transparent Interface of Augmented Glass Bottles

MusicBottles introduced a tangible interface where physical bottles serve as containers and controls for digital information (Photo 21.8). The system consists of a specially designed table and systems of corked bottles that "contain" the voices of contributing instruments. For example, a classical variation (containing Edouard Lalo's Piano Trio in C Minor,

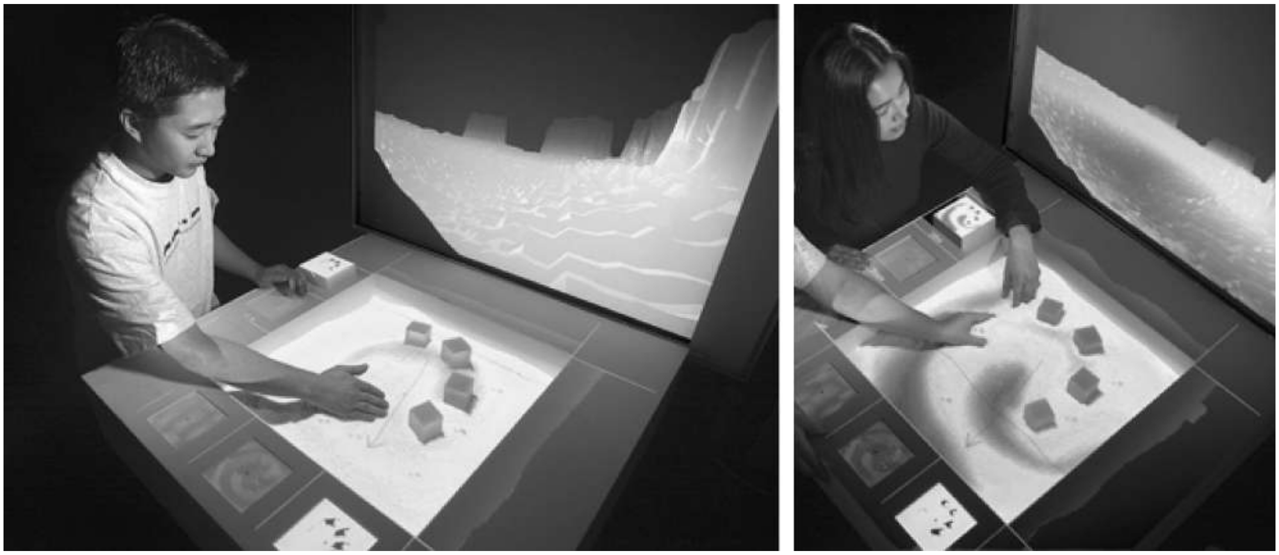


PHOTO 21.7 Using SandScape, users can alter the form of a landscape model by manipulating sand. The effects of computational analysis are generated and projected upon the sand's surface in realtime.

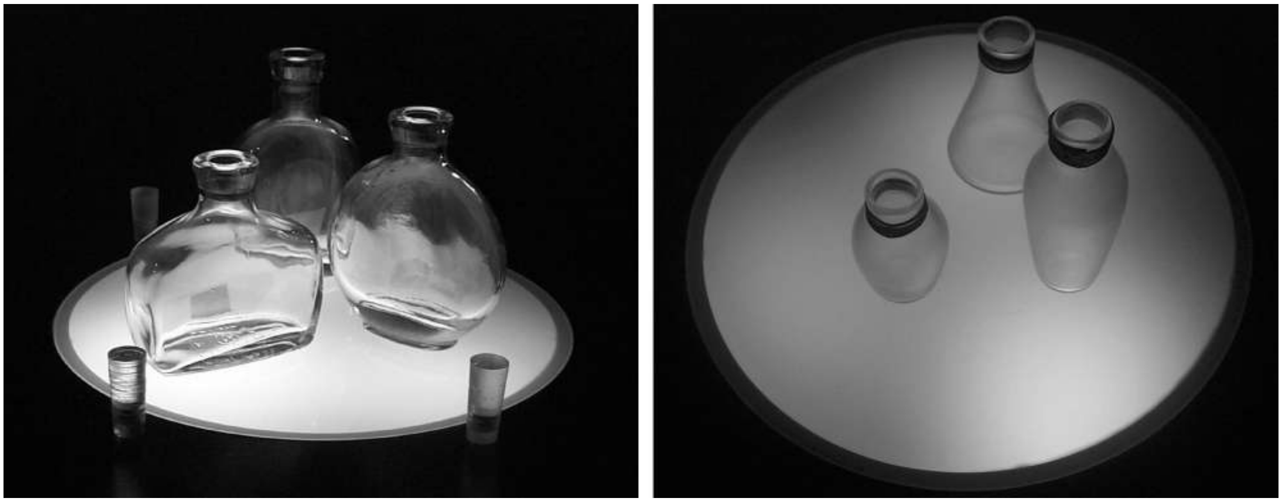


PHOTO 21.8 musicBottles. musicBottles is a tangible interface in which bottles exist as containers and controls for digital information. The system includes a specially designed table and (here) three corked bottles that "contain" the sounds of classical, jazz, and techno music. The interface allows users to structure the experience of the musical composition by physically manipulating the different sound tracks.

Op. 7) contained three bottles—violin, cello, and piano; bottle sets for jazz, electronic music, weather, and various spoken voice ensembles—were also realized. Custom electromagnetic tags embedded in the bottles and their stoppers enable each to be individually identified.

When a bottle is placed onto the stage of the instrumented table, the system identifies the bottle, and illuminates the stage to indicate the bottle has been recognized. The opening and closing of bottles is also detected; as the stopper is removed, the corresponding instrument becomes audible. A pattern of colored light is rear-projected onto the table's translucent surface to reflect changes in each instrument's pitch and volume. The interface allows users to structure the experience of the musical composition through physically manipulating the contributing voices.

Humans have used glass bottles for thousands of years. Through extending the affordances and metaphors of physical bottles within our interwoven physical and digital world, the bottles project explored new dimensions of interface transparency (Ishii 2004).

A wide variety of contents, including music, weather reports, poems, and stories were designed to investigate the concept (Ishii et al. 1999). The arrangement of many hand-blown musicBottles upon the purpose-designed table; the feel of the glass on opening; the music; and the light from the LED lamps, illuminating the sometimes diffuse, cracked, and colored glass together create a highly evocative aesthetic experience—one not found from the click of a mouse.

Potential applications are not limited to music. One might imagine perfume bottles filled with poetry, or wine bottles that decant stories (Mazalek, Wood, and Ishii 2001). More practical applications might include medicine chests of bottles that tell the user how and when to take them, and let the hospital know when they do. (This scenario has begun to be realized with Vitality's GlowCaps.) As an intimate part of our daily lives, augmented glass bottles offer a compelling instance and vision for simple, cognitively transparent tangible interfaces.

21.10.7 InTouch: Tangible Telepresence through Distributed Synchronized Physical Objects

InTouch explored new forms of interpersonal communication over distance through touch by preserving the physical analog movement of synchronized distributed rollers (Brave and Dahley 1997; Brave, Ishii, and Dahley 1998). Force feedback was used to create the illusion that people, separated by distance, are interacting with a shared physical object. The "shared" object provides a haptic link between geographically distributed users, opening a channel for physical expression over distance.

InTouch is a paired tangible, with each built around three rollers (Photo 21.3). Each roller is synchronized to the corresponding roller on the distant peer mechanism; when one roller is moved, the corresponding distant roller also moves. If the movement of one roller is held, the roller transmits that resistance to the other roller. The rollers are in a sense connected by a stiff computational spring. Two users separated by distance can then play, moving or tapping the rollers or more passively feel the other person's manipulation of the object. The presence of the other person is represented tangibly through physical interaction with the inTouch device.

Force feedback is commonly used to allow a user to "touch" virtual objects in the computer screen through a single point. Instead, inTouch applies this technology to realize a link for interpersonal haptic communication. InTouch allows people to feel as if they are connected through touching the rollers, to another person. Instead of touching inanimate objects, each person is touching a dynamic, shared physical object.

Some of the important features of inTouch from an HCI perspective include the following:

1. No boundary between "input" and "output" (input/output coincidence: the wooden rollers are force displays as well as input devices)
2. The principal instruments of human input and output are the hands, not the eyes/ears (with touch being the primary modality)
3. Information can be sent and received simultaneously through one's hands

The telephony and telepresence communities have framed their ultimate goal as reproducing human voice and imagery as realistically as possible, toward creating the illusion of "being there." In the spirit of "beyond being there" (Hollan and Stornetta 1992), inTouch takes the opposite approach, making users aware of the remote partner without rendering them in bodily terms, creating what we call a tangible or ghostly presence. By seeing and feeling tangibles physically moving with human nuance, we imagine the influence of ghostly bodies. This concept of ghostly presence provides us an alternate perspective—a minority report, as it were—on potential futures for telepresence.

21.10.8 Curlybot: A Toy to Record and Play

Curlybot is a toy that can record and replay physical motion (Photo 21.9). As one plays with the device, it remembers how it has been moved, and can replay this movement with all the intricacies of the original gesture. Every pause and acceleration, even shaking in the user's hand, is recorded. Curlybot then repeats this gesture indefinitely, creating beautiful and expressive patterns. Children can use curlybot to gain a strong intuition for advanced mathematical and computational concepts (e.g., differential geometry) through play in a context fully apart from traditional HCI (Frei et al. 2000).

The actuation technology developed for real-time communication in inTouch was used in curlybot for gestural recording and playback. Two motors equipped with an optical encoder enable free rotation, in addition to forward and backward movement.

PHOTO 21.9 Curlybot. Curlybot is a toy that records and plays back physical motion. Curlybot remembers how it has been moved by human users, and can replay this movement with all the intricacies of the original gesture. Every pause and acceleration – even shaking in users' hands – is recorded and repeatedly relived, allowing simple gestures to expand into complex geometric patterns.

When the user presses curlybot's lone button, a proximal LED is illuminated, indicating recording is in progress. The user then moves curlybot about; meanwhile, an encoder records this embodied gesture. Pushing the button a second time terminates recording; the LED passes from red to green, indicating playback is in progress. The internal microprocessor compares the current position with stored positions, and guides the motors toward retracing the steps within curlybot's memory.

This project contributes both to interface design and education. As a tangible interface, like inTouch, curlybot blurs the boundary between input and output. Curlybot is the indivisible sum of both an input device which record gestures, and a self-actuating display that re-enacts them. By allowing the user to teach curlybot gestures with her hand, then bodily re-enacting these gestures in physical space, curlybot enables a strong connection between body and mind—one going beyond any facsimile expressible on a computer screen.

From an educational standpoint, curlybot allows very young children to explore "advanced" mathematical and computational concepts. Curlybot supports new ways of thinking about geometric shape and pattern. Children can use curlybot to explore basic ideas behind computational procedures, such as how complexity can result from simple primitives. This is similar to outcomes possible with the Logo programming language, but does not require children to read or write, thus making powerful ideas experientially accessible to younger children. Curlybot also draws strongly on children's intuition about their own physical actions in the world toward the ends of learning—what Seymour Papert calls "body syntonic learning" (Papert 1980). Finally, the direct input and beautifully expressive patterns that result from curlybot's Mnemosynetic dance keep children playing and engaged.

21.10.9 Topobo: Three-Dimensional Constructive Assembly with Kinetic Memory

Topobo, from the terms "topology" and "robotics," is a three-dimensional (3D) constructive assembly system with kinetic memory (Raffle, Parkes, and Ishii 2004). Like curlybot, topobo has the ability to record and replay physical motion. By snapping together a combination of passive (static) and active (motorized) components, Topobo users can quickly assemble dynamic, biomorphic forms (e.g., resembling animals and skeletons). Topobo allows users to animate these forms by recording movement—for example, pushing, pulling, and twisting—and later observe the system cyclically replay these motions. This kinetic memory aspect is inherited from curlybot, while its constructive geometric language derives from the commercial toy Zoob.

As an example, a dog can be constructed, then taught to gesture and walk by twisting its body and legs. The dog will repeat these movements, walking indefinitely forward (as space allows). Similar to the ways people learn about static structures through passive building blocks, users can learn about dynamic structures through play with topobo. Topobo embeds computation within a dynamic building system, allowing gestural manipulation to become a kind of primitive programming language (Photo 21.10).

Topobo was inspired by current trends in computational media design, and by artists and empiricists using visual explorations and models of natural phenomena to more deeply appreciate patterns found within the natural world. In this spirit, Topobo allows people to use experimentation, play, and self-expression to discover and explore relationships between natural forms and dynamic motion.

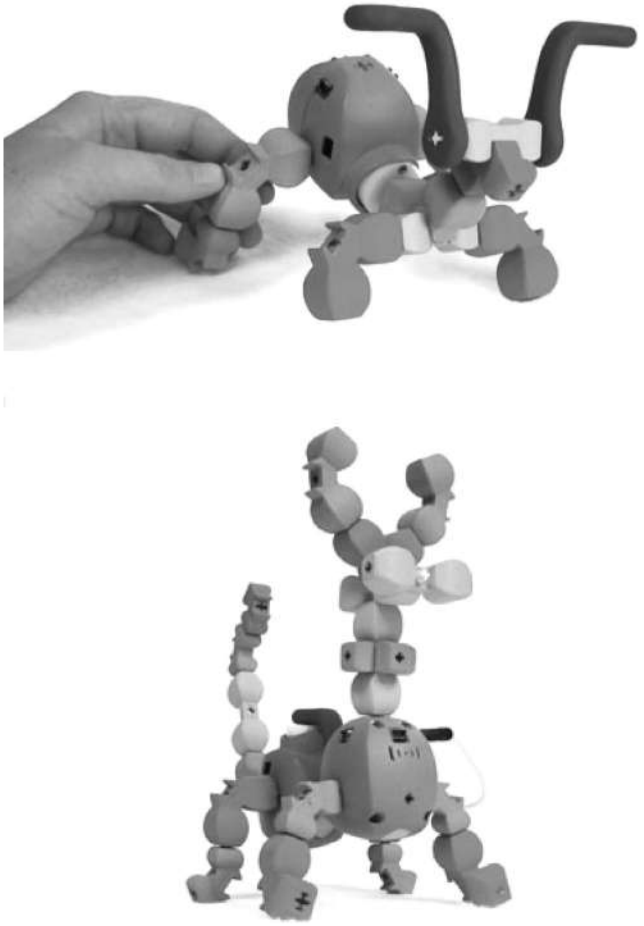


PHOTO 21.10 Topobo. Topobo is a 3D constructive assembly system with kinetic memory: the ability to record and play back physical motion. Topobo is distinguished by its coincident physical input and output behaviors.

21.10.10 MediaBlocks: A Token and Constraint-Based System

The mediaBlocks system is a tangible interface for manipulating lists of online digital media such as video clips and images (Ullmer, Ishii, and Glas 1998). Whereas Urp provides a spatial interface for leveraging object arrangements consistent with real-world building configurations, the mediaBlocks system provides a relational interface for manipulating more abstract digital information.

MediaBlocks are small, digitally tagged blocks, dynamically bound to elements or lists of online media. MediaBlocks support two major modes of use. First, they function as containers for physically embodying network-based (cloud) media, and for moving it between different media devices (e.g., conference room cameras, digital whiteboards, wall displays, and printers). In a sense, this role combines a tangible container with physically situated copy and paste.

Second, mediaBlocks can be used as physical containers and controls on a media sequencing device (Photo 21.11). Partially modeled after the tile racks of the board game Scrabble^{TM} a "sequence rack" allows the media contents

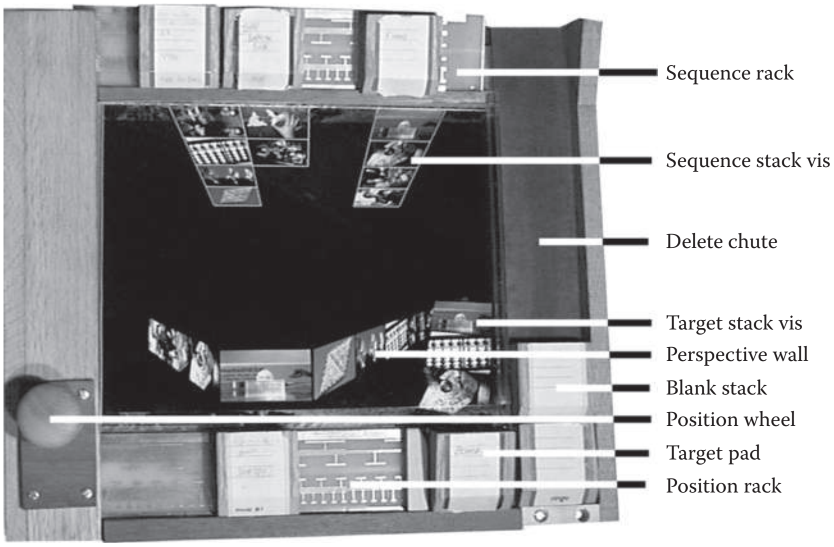


PHOTO 21.11 MediaBlocks: media sequencing device. The mediaBlocks system is a tangible interface for manipulating lists of online digital media (e.g., video clips and images). MediaBlocks act as both containers and controls, with behaviors determined by physical constraints within artifacts like this media sequencing device.

of multiple adjacent mediaBlocks to be copied into a new mediaBlock container. Similarly, a "position rack" maps the physical position of a block to an indexing operation upon its contents. When mediaBlocks are positioned on the left edge of the position rack, the first media element of the block is selected. Intermediate physical positions on the rack provide access to later elements in the associated media list of the block. Like its Marble Answering Machine, TuneTown, and LogJam predecessors (Crampton Smith 1995; Cohen, Withgott, and Piernot 1999; Singer et al. 1999), and descendants like Tagged Handles and Tangible Query Interfaces (MacLean, Snibbe, and Levin 2000; Ullmer, Ishii, and Jacob 2003), mediaBlocks illustrates how systems of physical tokens and constraints can embody and support manipulation of abstract digital information and operations.

21.10.11 Pinwheels: Ambient Interfaces to Digital Information

Pinwheels is an example of ambient media. They demonstrate a new approach to interfacing people with online digital information through subtle changes in sound and movement, perceptible in the background of awareness. Pinwheels spin in a "wind of bits," representing an invisible flow of digital information such as network traffic, embodied as physical movement within architectural spaces (Photo 21.12).

Nature is filled with subtle, beautiful, expressive ambient media that engage each of our senses. The sounds of rain and feelings of warm wind on our cheeks help us understand and enjoy the weather even as we engage in other activities. Similarly, we are aware of the activity of neighbors through passing sounds and shadows at the periphery of our attention. Cues like an open door or lights in an office help us subconsciously understand the activities of other people, and communicate our own activity and availability.

Current personal computing interfaces largely ignore these rich ambient spaces, and squeeze vast amounts of digital information into small rectangular screens. We seek to broaden the concept of "display," making use of the entire physical environment as an interface. Using ambient media, information can be manifested as subtle changes in form, movement, sound, color, smell, temperature, or light. We call such systems ambient displays.

PHOTO 21.12 Pinwheels. Pinwheels is an example of ambient information display. The flow of digital information is presented at the periphery of human perception.

Pinwheels evolved from the idea of using airflow as ambient media. However, we found the flow of air itself was both difficult to control, and challenging to interpret as a conveyance of information. As an alternative, we envisioned that a visual/physical representation of airflow based on spinning pinwheels could be both legible and poetic. Pinwheels spin in the "bit wind" with different speeds and dynamics based upon their evolving information source. For example, an atmospheric scientist might map patterns of solar wind into patterns of pinwheel spinning within a room.

We envisioned ambient displays as being sited at many locations in architectural space, and suited to the display of the following:

1. Human presence (awareness of physically and/or temporally remote human activities and status)
2. Atmospheric and astronomical phenomena
3. General states of large and complex systems (e.g., diverse physical and digital infrastructure)

There are many design challenges surrounding ambient displays. One is the mapping of information to physical motion and other ambient media. A designer of ambient displays must transform digital data into a meaningful pattern of physical motion that successfully communicates the source information. The threshold between foreground and background is another key issue. Ambient displays are expected to go largely unnoticed until some change in the display or of users' attention brings it into the attentional foreground. How to best keep the level of display at the threshold of attention remains an open design issue.

21.11 CONTRIBUTIONS OF TANGIBLE USER INTERFACES

Tangible interfaces have several important advantages over traditional graphical interfaces within certain contexts, as well as attendant limitations. This section summarizes and discusses some of these.

21.11.1 Double Interaction Loop—Immediate Tactile Feedback

One important advantage of TUI is that users receive passive haptic feedback from tangibles as they are grasped and manipulated. This allows users to complete actions kinesthetically without waiting for digital (primarily visual) feedback.

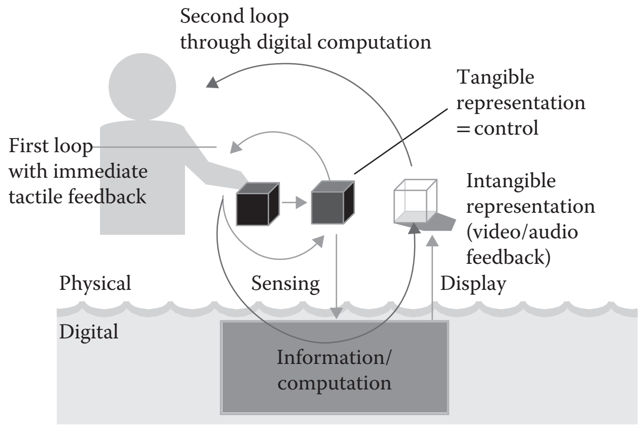


FIGURE 21.4 TUI's Double Feedback Loops. TUI provides two feedback loops: 1) immediate tactile feedback, and 2) feedback through digital processing (with possible delay).

Typically there are two feedback loops in TUI, as shown in Figure 21.4:

1. The passive haptic feedback loop provides users with immediate confirmation that they have grasped and moved the object. This loop exists within a physical domain and does not require computational sensing or processing. Thus, there is no computational delay. The user can begin manipulating tangibles as desired without needing to wait for the second feedback loop: mediated (typically visual) confirmation from the interface. In contrast, when one uses a mouse with a GUI, he or she must wait for visual feedback (the second loop) to complete an action.
2. The second circle is a digital feedback loop that requires sensing tangibles as moved by users, computing responses to the sensed data, and displaying the results as visual (and auditory) feedback. Thus, this second loop takes longer than the first.

Many frustrations from using current computers come from the delay of digital feedback, as well as weak confirmations of computational actions. We believe the double loops of TUI provide users a path toward easing these frustrations.

Note: Actuation technology, as illustrated by the Actuated Workbench, will contribute to add another loop. Figure 21.5 illustrates the third loop introduced into the TUI model by computer-controlled actuation and sensing. The third loop allows the computer to give feedback on the status of the digital information as the model changes or responds to internal computation.

21.11.2 Persistence of Tangibles

As physical artifacts, tangibles are persistent. Tangibles also maintain physical state, with their physical configurations typically tightly coupled to the digital state of the systems they represent. A distinction between in-band and out-of-band interactions is also an important consideration (Ullmer, Ishii, and Jacob 2005).

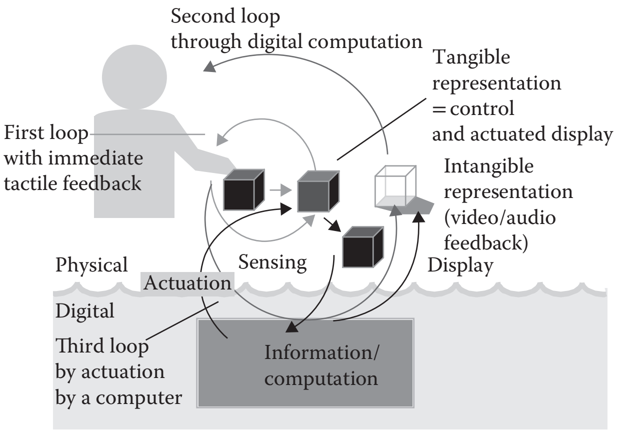


FIGURE 21.5 TUI with actuation (actuated workbench). Computational actuation provides another loop for computer to control position of objects (tangible representations) on the workbench.

Tangibles' physical state embodies key aspects of the underlying computation's digital state. For example, the identity, position, and orientation of Urp's tangibles upon the workbench serve central roles in representing and controlling the state of the underling digital simulation. Even if the mediating computers, cameras, and projectors of Urp are turned off mid-interaction many aspects of the system's state remain concretely expressed by its physical configuration. In contrast, the physical form of the mouse holds little representational significance, as GUIs represent information almost entirely in visual form.

21.11.3 Coincidence of Input and Output Spaces

Another important feature and design principle of TUI is coincidence of input and output spaces. This supports the provision of seamless information representations spanning both tangible (physical) and intangible (digital) domains.

GUIs prototypically utilize the mouse and keyboard as generic "remote" controllers (input), with the screen serving as the primary output medium. Thus, a spatial discontinuity separates these two spaces. (Some evolution has begun with increasing popularity of touch interaction with smartphones, tablets, et al.) There is also multimodal inconsistency, as touch is the main input while vision is the primary output.

TUIs work to coincide input and output spaces (with some exceptions) to realize a seamless coupling of physical and digital worlds (Ishii and Ullmer 1997). Urp has provided one strong example (Underkoffler and Ishii 1999). Illuminating Clay (Piper, Ratti, and Ishii 2002) and SandScape (Ishii et al. 2004) provide two other examples of input/output coincidence using continuous flexible materials (sand). Curlybot and topobo also demonstrate this concept, using actuated tangibles simultaneously as input and output.

21.11.4 Special Purpose versus General Purpose

From an interaction perspective, GUIs are deeply general-purpose interfaces that mediate varied applications visually using dynamic pixels and generic remote controllers. In comparison, TUIs are relatively specific interfaces tailored to certain types of applications in order to increase the directness and intuitiveness of interaction, facilitate multi-user interaction, and support expanded modalities for maintaining awareness.

In Urp, a notable design decision is its use of tangibles with very application-specific physical forms (e.g., partially abstracted building models and clock representations) as fundamental elements of the interface. These give users important visual and tactile information about the conceptual and computational entities they represent. Some tangibles may invert prior artifact behaviors. For example, in Urp, instead of the clock hands moving to indicate the passage of time, the user manipulates the clock to change time of day for shadow studies (Photo 21.1). Similarly, she can change the orientation of the weather vane to control the direction of the wind (Photo 21.2).

Of course, this special-purposeness of TUIs can become a disadvantage if users wish to apply them to widely diverse applications, as tangibles tailored to certain application may not be reusable in others. By making the form of objects more abstract (e.g., a round puck), one loses the legibility of tangible representation. Here, the tangibles may transform into generic handles, rather than a physically legible representation of underlying digital information. Which design choice is preferred varies between systems, users, and contexts, and remains both art and science in equal measure. Altogether, a balance between specific/concrete versus generic/abstract forms, sometimes in the form of core versus domain tangibles (Ullmer et al. 2008; Ullmer et al. 2010), seems an important component of good design.

21.11.5 Space-Multiplexed Input

Another feature and contribution of tangible interfaces concerns space-multiplexed input (Fitzmaurice, Ishii, and Buxton 1995a). Each tangible representation serves as a controller occupying its own space. This encourages two handed and multiuser interaction with underlying computational models. In this way, TUIs are often particularly suitable for collocated collaboration, allowing concurrent manipulation of information by multiple users.

In contrast, GUIs traditionally have provided time-multiplexed input, allowing users to use one generic device to control different computational functions at different points in time. For instance, the mouse is used for menu selection, scrolling windows, pointing and clicking buttons in a time-sequential manner.

TUI can support not only collocated collaboration, but also remote collaboration. Here, actuation can become important for synchronizing the physical states of tangibles over distance (Pangaro et al. 2002). Combining the Urp scenario with the Actuated Workbench, it is possible to have multiple distributed Urp tables in different locations, linked and synchronized over the Internet. One Urp can be in Tokyo and another in Boston, with the building tangibles and their shadows synchronized as the distributed planning team collaborate through the system. This synchronization allows both teams to discuss and effect changes in real-time, providing a common reference for otherwise ethereal qualities such as wind, time, and shadows.

21.12 DESIGN CHALLENGES AND OPPORTUNITIES

Tangible interfaces are today the subject of much activity—both in academic research labs, niche contexts such as museums and classrooms, and broader domestic and commercial use. In parallel, many important design issues are still at best partially understood. We identify and discuss several of these, partly toward fostering continuing academic study and commercial engagement.

- What should be tangible, and what should be intangible?

  Not all elements of computational systems can, or should, be physically embodied. Choosing which system elements should be tangible is a critical and still challenging decision. One heuristic is to physically embody the "key objects of interest" (Ullmer 2002; Hornecker and Buur 2006). Tangibility decisions also relate to many aspects of application context. Are eyes, ears, or hands typically engaged by other aspects of the system or surrounding context (Cohen and Oviatt 1995)? Is mobility important? Does the application typically engage one or multiple users? If multiple, are they collocated or distributed, peers or differentiated?

- Does a given physical artifact have a digital binding?

  Not all physical artifacts are, or should be, bound to digital associations. Today, most physical artifacts have no digital associations. How should users distinguish tangibles from "ordinary objects?" Sometimes, physical and temporal context can provide sufficient cues. For example, the box of a stored board game helps segregate playing elements from "other stuff"; similar approaches can be used for TUIs. Specialized physical or visual design, potentially including special human and/or machine-parseable symbols or conventions, can help in this differentiation (Ullmer et al. 2010). In other cases, mediation or engagement with the artifact may be required.

- What is a tangible's digital binding (both in the absence and presence of mediation and interaction)?

  Tangible interfaces use widely varying modes of representation. Some use one or a few relatively generic tangibles, leaving most representation to computational mediation. Some incorporate static text and other visual content; others use evocative physical forms with little or no "skin" differentiation. Some incorporate dynamic visual displays (e.g., LEDs or bistable displays like e-ink) and/or actuators.

These varying representational forms may be sufficient for novice or expert users to unambiguously identify a tangible's bindings—or not. "In-band" engagement (e.g., with a mediating surface or glasses) may be required, as may "out-of-band" activities—consulting a supporting GUI, manual, or expert user. In time, codified "human interface guidelines" (as championed by Apple and other companies) may help users evolve specific expectations. Trust is also at stake. Users are often unaware of important activities web pages and mobile phones are taking on behalf of them (or others). Similar challenges likely lie ahead for tangibles.

- How might tangibles best coexist and engage with other genres of HCI?

It is unlikely that tangible interfaces will "obsolete" graphical interfaces, speech interfaces, or other interaction modalities. We have argued that tangible interfaces are a strong fit for a variety of application contexts. Still, it is unlikely that (say) the full spectrum of GUI office productivity applications would evolve into tangible form. Also, hybrid interfaces are a distinct possibility. For example, while some interfaces are "purely" graphical or textual, hybrid combinations of graphical and textual interaction techniques are common. Analogies may likely unfold for tangible and graphical interaction, as well as between TUIs and other post-WIMP techniques (Jacob et al. 2008)—for example, tangible augmented reality (Lee et al. 2004). This presents both needs and opportunities for physical and graphical approaches that bridge multiple paradigms.

Another opportunity relates to the progressive realization of Weiser's ubiquitous computing vision. It is now routine for users to wear and carry several "computers" (e.g., laptop, slate, handheld, glasses, and garments), amidst many others deployed on desks, walls, ceilings, and floors. With wireless networking near-ubiquitous in some regions, data interchange is clearly technically possible. However, paradigms for compelling coordination of these complex ecologies, joined by tangibles and other interactives, arguably remain in infancy. With their native physicality, tangibles hold a potentially privileged role in helping mediate these physical/digital ecosystems. Helping manage scarce attention is one important component; comediating collaborative work is another.

- In a world where tangibles are common, how are they likely to be made?

New graphical interfaces can spring fully realized upon screens at the speed of electrons and photons. Tangibles experience a more labored gestation and delivery. Paper printing, electronic device fabrication, and handcrafts each lend valuable perspective on possible futures.

At least four locales for paper printing are now common. Much paper is bulk-printed in distant lands, delivered in days or weeks. Stores in the same city or neighborhood can offer similar services in hours, typically trading cost for service and customization. Many homes and offices now have multiple printers per room, bringing renditions in minutes or seconds. Bistable paper displays can trim delivery to milliseconds.

A similar story likely suggests one path in the future of tangibles. Diverse 2D, 3D, and higher-dimensional printers and cutters—some allowing printed artifacts to "walk out" by their own volition (Gershenfeld 2005)—are becoming common at each of these physical locales. Tangibles today and tomorrow are finding origin in each of these settings, each bringing different trade-offs.

The fabrication of electronic devices and handcrafts provide different perspectives. Today, most electronic devices are mass-produced far from their point of consumption, driven by high fixed costs and low wages. Handcrafts are created both at global and local origins—in both cases, again typically earning meager returns for the laborer. These different production pathways also have consequential implications for sustainability (Blevis 2007; Kloepffer 2008; DiSalvo, Sengers, and Brynjarsdottir 2010; Geyer and Doctori Blass 2010) and the design and engineering approaches by which tangible interfaces are realized (Greenberg and Fitchett 2001; Beigl and Gellersen 2003; Villar and Gellersen 2007; Buechley et al. 2008; Sankaran et al. 2009).

The mass production and paper printing models are likely to suggest the most common paths for fabricating tangibles. However, in a time where myriad communities and whole continents struggle desperately toward finding a future, the physicality, material craft, and potential cultural specificity of tangibles potentially offer real and promising prospects.

21.13 CONCLUSION

Ishii met a highly successful computational device called the "abacus" when he was 2 years old (Photo 21.13). He could enjoy the touch and feel of the "digits" physically represented as arrays of beads. This simple abacus was not merely a digital computational device. Because of its physical affordances, the abacus also became a musical instrument, imaginary toy

PHOTO 21.13 Abacus. The beads, rods, and frame of abacus serve as physical representations of both information (numbers) and computation (arithmetic operations). They serve as directly manipulable physical controls to compute on numbers. The abacus can be seen as a model of tangible interfaces.

train, and a back scratcher. He was captivated by the sound and tactile interaction with this simple artifact.

His childhood abacus also became a medium of awareness. When his mother kept household accounts, he was aware of her activities by the sound of her abacus, knowing he could not ask to play while her abacus made its music.

This abacus suggests to us a new direction of HCI that we call TUI. The abacus makes no distinction between "input" and "output." Rather, the beads, rods, and frame serve as physical representations of numerical information and computational mechanism. They also serve as directly manipulable physical controls to computationally engage with numbers.

Also, the simple, transparent mechanical structure of the abacus (absent any digital black boxes) provides rich physical affordances (Norman 1999). Even children can understand many potentials of this artifact without reading a manual.

Tangible interfaces pursue these features further into the digital domain by giving physical form to digital information and computation, using physical artifacts both as representations and controls for computational media. The TUI design challenge concerns seamless extension of the physical affordances of artifacts into the digital domain.

This chapter has introduced basic concepts and example applications to address key properties and design challenges of tangible interfaces. The TUI approach still remains in its infancy, and extensive research is required to identify the "killer applications," scalable methods for their realization, and a set of strong design principles.

Tangible interface research naturally and fruitfully intersects with the paths of industrial/product design, as well as environmental design, architecture, and interior architecture (Kurtich and Eakin 1993; Kernaghan 2011). TUIs have made an impact on the media arts/interactive arts community. The authors hope tangible interfaces will continue promoting these interdisciplinary design research movements within and beyond the HCI community, bringing strong design culture and a media arts perspective both to the scientific/academic communities, and across the far broader, diversely populated ecologies of our beautiful, challenged planet.

Mark Weiser's seminal article on Ubiquitous Computing (Weiser 1991) began with the paragraph:

> The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.

We believe tangible interfaces are one of the promising paths toward Weiser's vision.

ACKNOWLEDGMENTS

The authors thank many of the former and current Tangible Media Group (TMG) students and colleagues in the MIT Media Lab for their contributions to tangible interface research. Special thanks are due to Dr. John Underkoffler, Dr. Kimiko Ryokai, Dr. James Patten, Dr. Angela Chang, Dr. Hayes Raffle, Dr. Amanda Parkes, Dr. Oren Zuckerman, Dr. Alexandra Mazalek, Matthew Gorbet, Paul Yarin, Jamie Zigelbaum, and many others. Thanks are also due to Things That Think and Digital Life Consortia at the Media Lab for support of our research. Ullmer thanks the students and postdoctoral researchers of LSU's Tangible Visualization group, the faculty and staff of LSU CCT and the Department of Computer Science, and the NSF MRI and Creative-IT programs for support. Ishii thanks Professor Bill Buxton and Dr. George Fitzmaurice for collaboration on the Bricks project in 1994 at University of Toronto (Fitzmaurice, Ishii, and Buxton 1995b) which laid the foundation of TUI, and Professor Mitchel Resnick for his vision of Digital Manipulatives (Resnick et al. 1998), which influenced our design of educational TUI applications.

REFERENCES

Aish, R. 1979. 3D input for CAAD systems. *Comput Aided Des* 11(2):66–70.

Aish, R., and P. Noakes. 1979. Architecture without numbers—CAAD based on a 3D modelling system. *Comput Aided Des* 16(6):321–28, Nov. 1984.

Anagnostou, G., D. Dewey, and A. Patera. 1989. Geometry-defining processors for engineering design and analysis. *Visual Comput* 5:304–15.

Anderson, D., J. L. Frankel, J. Marks, A. Agarwala, P. Beardsley, J. Hodgins et al. 2000. Tangible interaction + graphical interpretation: A new approach to 3D modeling. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, 393–402. New York: ACM.

Arias, E., H. Eden, and G. Fischer. 1997. Enhancing communication, facilitating shared understanding, and creating better artifacts by integrating physical and computational media for design. In *Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 1–12. New York: ACM.

Astrahan, M., B. Housman, J. Jacobs, R. Mayer, and W. Thomas. 1957. Logical design of the digital computer for the SAGE system. *IBM J Res Dev* 1(1):76–83.

Baskinger, M., and M. Gross. 2010. Tangible interaction = form + computing. *ACM interact* 17(1):6–11.

Baudisch, P., T. Becker, and F. Rudeck. 2010. Lumino: Tangible blocks for tabletop computers based on glass fiber bundles. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, 1165–74. New York: ACM.

Beigl, M., and H. Gellersen. 2003. Smart-its: An embedded platform for smart objects. In *Smart Objects Conference (sOc)*. Paris: France Telecom and French CNRS.

Ben-Joseph, E., H. Ishii, J. Underkoffler, B. Piper, and L. Yeung. 2001. Urban simulation and the luminous planning table: Bridging the gap between the digital and the tangible. *J Plann Educ Res* 21(2):196–203.

Benhammous, D. 2002. *Large Scale Electrostatic Actuation*. B.S. Mechanical Engineering, MIT.

Bennett, P. 2010. The representation and control of time in tangible user interfaces. *Proc. of TEI '10*, 307–8. New York: ACM.

Biegelsen, D., A. Berlin, P. Cheung, M. Fromherz, D. Goldberg, W. Jackson et al. 2000. AirJet paper mover. In *SPIE Int. Symposium on Micromachining and Microfabrication*, 4176–11. Bellingham, WA: SPIE.

Blackwell, A. F. 2003. Cognitive dimensions of tangible programming languages. In *Proc. of PPIG '03*, 391–405.

Blevis, E. 2007. Sustainable interaction design: invention & disposal, renewal & reuse. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 503–12. New York: ACM.

Brave, S., and A. Dahley. 1997. inTouch: A medium for haptic interpersonal communication (short paper). In *Conference on Human Factors in Computing Systems (CHI '97).* New York: ACM.

Brave, S., H. Ishii, and A. Dahley. 1998. Tangible interfaces for remote collaboration and communication. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW '98)*, 169–78. New York: ACM.

Buechley, L., M. Eisenberg, J. Catchen, and A. Crockett. 2008. The LilyPad Arduino: Using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 423–32. New York: ACM.

Burbeck, S. (Producer). 1992. Applications Programming in Smalltalk-80(TM): How to use Model-View-Controller (MVC). http://st-www.cs.uiuc.edu/users/smarch/st-docs/mvc.html.

Buxton, W. 1995. Integrating the periphery and context: A new model of telematics. *Graphics Interface '95*, 239–46. New York: ACM.

Chang, A., S. O'Modhrain, R. Jacob, E. Gunther, and H. Ishii. 2002. ComTouch: Design of a vibrotactile communication device. In *Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 312–20. New York: ACM.

Chang, A., B. Resner, B. Koerner, X. Wang, and H. Ishii. 2001. LumiTouch: An emotional communication device. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 313–14. New York: ACM.

Cheng, K.-Y., R.-H. Liang, B.-Y. Chen, R.-H. Laing, and S.-Y. Kuo. 2010. iCon: Utilizing everyday objects as additional, auxiliary and instant tabletop controllers. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, 1155–64. New York: ACM.

Cohen, P. R., and S. L. Oviatt. 1995. The role of voice input for human-machine communication. In *Proceedings of the National Academy of Science*, 92(22):9921–27. Washington, DC: National Academy of Sciences.

Cohen, J., M. Withgott, and P. Piernot. 1999. LogJam: A tangible multi-person interface for video logging. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*, 128–35. New York: ACM.

Couture, N., G. Riviere, and P. Reuter. 2008. GeoTUI: A tangible user interface for geoscience. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*, 89–96. New York: ACM.

Crampton Smith, G. 1995. The Hand That Rocks the Cradle. *I.D.*, 60–65.

Csikszentmihaly, M., and E. Rochberg-Halton. 1981. *The Meaning of Things: Domestic Symbols and the Self*. Cambridge, UK: Cambridge University Press.

Dahley, A., C. Wisneski, and H. Ishii. 1998. Water lamp and pinwheels: Ambient projection of digital information into architectural space (short paper). In *Conference on Human Factors in Computing Systems (CHI '98)*, Conference Summary of CHI '98. New York: ACM.

DiSalvo, C., P. Sengers, and H. Brynjarsdottir. 2010. Mapping the landscape of sustainable HCI. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, 1975–84. New York: ACM.

Djajadiningrat, T., S. Wensveen, J. Frens, and K. Overbeeke. 2004. Tangible products: Redressing the balance between appearance and action. *Pers Ubiquitous Comput* 8(5):294–309.

Dodge, C. 1997. The bed: A medium for intimate communication. In *CHI '97 Extended Abstracts on Human Factors in Computing Systems: Looking to the Future*, 371–72. New York: ACM.

Donath, J., and D. Boyd. 2004. Public displays of connection. *BT Technol J* 22(4):71–82.

Dourish, P. 2002. *Where the Action is: The Foundations of Embodied Interaction*. Cambridge, MA, Mass: MIT Press.

Dunne, A., and F. Raby. 1994. Fields and thresholds. *Presentation at the Doors of Perception 2*. http://www.mediamatic.nl/-Doors/Doors2/DunRab/--DunRab-Doors2-E.html.

Eden, H., E. Scharff, and E. Hornecker. 2002. Multilevel design and role play: Experiences in assessing support for neighborhood participation in design. In *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 387–92. New York: ACM.

Edge, D., and A. Blackwell. 2006. Correlates of the cognitive dimensions for tangible user interface. *J Visual Languages Comput* 17(4):366–94.

Etter, R., and C. Roecker. 2007. A tangible user interface for multi-user awareness systems. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 11–2. New York: ACM.

Everitt, K. M., S. R. Klemmer, R. Lee, and J. A. Landay. 2003. Two worlds apart: Bridging the gap between physical and virtual media for distributed design collaboration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 553–60. New York: ACM.

Fernaeus, Y., J. Tholander, and M. Jonsson. 2008. Towards a new set of ideals: Consequences of the practice turn in tangible interaction. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*, 223–30. New York: ACM.

Fischer, P. T., C. Zoellner, T. Hoffmann, and S. Piatza. 2010. VR/Urban: SMSingshot. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction*, 381–82. New York: ACM.

Fishkin, K. P. 2004. A taxonomy for and analysis of tangible interfaces. *Pers Ubiquitous Comput* 8:347–58.

Fitzmaurice, G. W. 1996. *Graspable user interfaces*.

Fitzmaurice, G. W., H. Ishii, and W. Buxton. 1995a. Bricks: Laying the Foundations for Graspable User Interfaces. In *Conference on Human Factors in Computing Systems (CHI '95)*, 442–9. New York: ACM.

Fitzmaurice, G. W., H. Ishii, and W. A. S. Buxton. 1995b. Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 442–9. New York: ACM.

Fogg, B., L. D. Cutler, P. Arnold, and C. Eisbach. 1998. HandJive: A device for interpersonal haptic entertainment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 57–64. New York: ACM.

Frazer, J. 1994. *An Evolutionary Architecture*, London, UK: Architectural Association.

Frazer, J., J. Frazer, and P. Frazer. 1980. Intelligent physical three dimensional modelling system. *Comput Graphics* 80:359–70.

Frei, P., V. Su, B. Mikhak, and H. Ishii. 2000. Curlybot: Designing a new class of computational toys. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2000)*, 129–36. New York: ACM.

Frey, M. 2007. CabBoots: Shoes with integrated guidance system. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 245–46. New York: ACM.

Gershenfeld, N. A. 2005. Fab: The coming revolution on your desktop—from personal computers to personal fabrication.

Geyer, R., and V. Doctori Blass. 2010. The economics of cell phone reuse and recycling. *Int J Adv Manuf Technol* 47(5):515–25.

Goldberg, A. 1984. *Smalltalk-80: The Interactive Programming Environment*. Reading, MA: Addison-Wesley.

Goldsmith, S. 1983. The readymades of marcel duchamp: The ambiguities of an aesthetic revolution. *J Aesthetics Art Criticism*, 42(2):197–208.

Gorbet, M., M. Orth, and H. Ishii. 1998. Triangles: Tangible interface for manipulation and exploration of digital information topography. In *Paper Presented at the Conference on Human Factors in Computing Systems (CHI '98)*, Los Angeles, CA. New York: ACM.

Greenberg, S., and C. Fitchett. 2001. Phidgets: Easy development of physical interfaces through physical widgets. In *Paper Presented at the Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, Orlando, FL. New York: ACM.

Hall, E. T. 1969. *The Hidden Dimension*. Garden City, NY: Doubleday.

Hinckley, K., R. Pausch, J. C. Goble, and N. F. Kassell. 1994. Passive real-world interface props for neurosurgical visualization. In *Conference on Human Factors in Computing Systems (CHI '94)*, 452–58. New York: ACM.

Hindus, D., S. D. Mainwaring, N. Leduc, A. E. Hagstrom, and O. Bayley. 2001. Casablanca: Designing social communication devices for the home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 325–32. New York: ACM.

Hines, J., T. Malone, P. Gonçalves, G. Herman, J. Quimby, M. Murphy-Hoye et al. 2011. Construction by replacement: a new approach to simulation modeling. *System Dynamics Review* 27:64–90.

Hollan, J., and S. Stornetta. 1992. Beyond being there. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 119–25. New York: ACM.

Holmquist, L. E., J. Redstr, and P. Ljungstrand. 1999. Token-based acces to digital information. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, 234–45. Berlin: Springer.

Holmquist, L. E., A. Schmidt, and B. Ullmer. 2004. Tangible interfaces in perspective: Guest editors\&rsquo; introduction. *Pers Ubiquitous Comput* 8(5):291–93.

Holmquist, L. E., and T. Skog. 2003. Informative art: Information visualization in everyday environments. In *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, 229–35. New York: ACM.

Hornecker, E., and J. Buur. 2006. Getting a grip on tangible interaction: A framework on physical space and social interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 437–46. New York: ACM.

Hummels, C., and A. V. D. Helm. 2004. ISH and the search for resonant tangible interaction. *Pers Ubiquitous Comput* 8(5):385–88.

Hurtienne, J., and J. H. Israel. 2007. Image schemas and their metaphorical extensions: Intuitive patterns for tangible interaction. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 127–34. New York: ACM.

Hutchins, E. 1995. *Cognition in the Wild*. Cambridge, MA: MIT Press.

Ishii, H. 2004. Bottles: A transparent interface as a tribute to Mark Weiser. *IEICE Trans Inf Syst* E87-D(6):1299–311.

Ishii, H., H. R. Fletcher, J. Lee, S. Choo, J. Berzowska, C. Wisneski et al. 1999. musicBottles. *ACM SIGGRAPH '99 Conference Abstracts and Applications*, 174. New York: ACM.

Ishii, H., M. Kobayashi, and K. Arita. 1994. Iterative Design of Seamless Collaboration Media. *Commun ACM (CACM)*, 37(8):83–97.

Ishii, H., A. Mazalek, and J. Lee. 2001. Bottles as a minimal interface to access digital information. *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 187–88.

Ishii, H., C. Ratti, B. Piper, Y. Wang, A. Biderman, and E. Ben-Joseph. 2004. Bringing clay and sand into digital design—continuous tangible user interfaces. *BT Technol J* 22(4):287–99.

Ishii, H., and B. Ullmer. 1997. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Conference on Human Factors in Computing Systems (CHI '97)*, 234–41. New York: ACM.

Ishii, H., C. Wisneski, S. Brave, A. Dahley, M. Gorbet, B. Ullmer et al. 1998. ambientROOM: Integrating ambient media with architectural space (video). In *Conference on Human Factors in Computing Systems (CHI '98)*, Conference Summary of CHI '98. New York: ACM.

Jacob, R. J. K., A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey et al. 2008. Reality-based interaction: A framework for post-WIMP interfaces. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 201–10. New York: ACM.

Jacob, R. J. K., H. Ishii, G. Pangaro, and J. Patten. 2002. A tangible interface for organizing information using a grid. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*, 339–46. New York: ACM.

Johnson, J., T. L. Roberts, W. Verplank, D. C. Smith, C. H. Irby, and M. Beard et al. 1989. The Xerox Star: A retrospective. *IEEE Comput* 22(9):11–25.

Jorda, S. 2008. On stage: The reactable and other musical tangibles go real. *Int J Arts Technol* 1(3–4):268–87.

Jorda, S., G. Geiger, M. Alonso, and M. Kaltenbrunner. 2007. The reacTable: Exploring the synergy between live music performance and tabletop tangible interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 139–46. New York: ACM.

Jouppi, N. P., S. Iyer, W. Mack, A. Slayden, and S. Thomas. 2004. A first generation mutually-immersive mobile telepresence surrogate with automatic backtracking. In *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, Vol. 1672, 1670–75. Washington, DC: IEEE.

Kalanithi, J. J., and V. Michael Bove Jr. 2008. Connectibles: Tangible social networks. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*, 199–206. New York: ACM.

Kernaghan, B. 2010. *Interiority: An Anthology of Critical Writing on Interior Architecture and Design*. London, UK: Architectural Press.

Kitamura, Y., Y. Itoh, and F. Kishino. 2001. Real-time 3D interaction with ActiveCube. *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 355–56. New York: ACM.

Klemmer, S. R., B. Hartmann, and L. Takayama. 2006. How bodies matter: Five themes for interaction design. In *Proceedings of the 6th Conference on Designing Interactive Systems*, 140–49. New York: ACM.

Klemmer, S., M. Newman, R. Farrell, M. Bilezikjian, and J. Landay. 2001. The designers' outpost: A tangible interface for collaborative web site. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, 1–10. New York: ACM.

Klemmer, S. R., M. Thomsen, E. Phelps-Goodman, R. Lee, and J. A. Landay. 2002. Where do web sites come from?: Capturing and interacting with design history. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*, 1–8. New York: ACM.

Kloepffer, W. 2008. Life cycle sustainability assessment of products. *Int J Life Cycle Assess* 13(2):89–95.

Kobayashi, K., M. Hirano, A. Narita, and H. Ishii. 2003. A tangible interface for IP network simulation. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, 800–01. New York: ACM.

Koleva, B., S. Benford, K. H. Ng, and T. Rodden. 2003. A framework for tangible user interfaces. In *Physical Interaction'03.*

Kraut, R., J. Galegher, and C. Egido. 1988. Relationships and tasks in scientific research collaboration. *SIGCHI Bull* 20(1):79–80.

Kurtich, J., and G. Eakin. 1993. *Interior Architecture*. New York, NY: Van Nostrand Reinhold.

Kuzuoka, H., and S. Greenberg. 1999. Mediating awareness and communication through digital but physical surrogates. In *CHI '99 Extended Abstracts on Human Factors in Computing Systems*, 11–12. New York: ACM.

Lawson, B. 2001. *The Language of Space*. Oxford and Boston: Architectural Press.

Lee, G. A., C. Nelles, M. Billinghurst, and G. J. Kim. 2004. Immersive authoring of tangible augmented reality applications. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, 172–81. New York: ACM.

Lee, J., V. Su, S. Ren, and H. Ishii. 2000. HandSCAPE: A vectorizing tape measure for on-site measuring applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 137–44. New York: ACM.

Levisohn, A., J. Cochrane, D. Gromala, and J. Seo. 2007. The meatbook: Tangible and visceral interaction. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 91–2. New York: ACM.

Lin, R.-T. 2007. Transforming taiwan aboriginal cultural features into modern product design: A case study of a cross-cultural product design model. *Int J Des* 1(2):45-53.

Ljungstrand, P., J. Redström, and L. E. Holmquist. 2000. WebStickers: Using physical tokens to access, manage and share bookmarks to the web. In *Proceedings of DARE 2000 on Designing Augmented Reality Environments*, 21–35. New York: ACM.

Mackay, W. E., and M. Beaudouin-Lafon. 2005. FamilyNet: A tangible interface for managing intimate social networks. In *Proc. of SOUPS '05.* New York: ACM.

MacLean, K. E., S. S. Snibbe, and G. Levin. 2000. Tagged handles: Merging discrete and continuous manual control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 225–32. New York: ACM.

Mankoff, J., A. K. Dey, G. Hsieh, J. Kientz, S. Lederer, and M. Ames. 2003. Heuristic evaluation of ambient displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 169–76. New York: ACM.

Mazalek, A., and E. van den Hoven. 2009. Framing tangible interaction frameworks. *AI EDAM*, 23(Special Issue 03): 225–35.

Mazalek, A., A. Wood, and H. Ishii. 2001. genieBottles: An interactive narrative in bottles. In *Conference Abstracts and Applications of SIGGRAPH '01*, 189. New York: ACM.

McNerney, T. S. 2004. From turtles to tangible programming bricks: Explorations in physical language design. *Pers Ubiquitous Comput* 8(5):326–37.

Merrill, D., J. Kalanithi, and P. Maes. 2007. Siftables: Towards sensor network user interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 75–8. New York: ACM.

Moggridge, B. 2007. *Designing Interactions*. Cambridge, MA: Mass: MIT Press.

Moran, T. P., E. Saund, W. V. Melle, A. U. Gujar, K. P. Fishkin, and B. L. Harrison. 1999. Design and technology for collaborage: Collaborative collages of information on physical walls. In *Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology*, 197–206. New York: ACM.

Mynatt, E. D. 2000. Co-opting everyday objects. In *Proceedings of DARE 2000 on Designing Augmented Reality Environments*, 145–46. New York: ACM.

Norman, D. A. 1999. Affordance, conventions, and design. *Interactions* 6:38–43.

Oba, H. 1990. *Environment Audio System for the Future*. Sony concept video.

Olson, G. M., and J. S. Olson. 2000. Distance matters. *Hum Comput Interact* 15: 139–79.

Pangaro, G., D. Maynes-Aminzade, and H. Ishii. 2002. The actuated workbench: Computer-controlled actuation in tabletop tangible interfaces. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST 2002)*, 181–90. New York: ACM.

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Patten, J., L. Griffith, and H. Ishii. 2000. A tangible interface for controlling robotic toys. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems*, 277–78. New York: ACM.

Patten, J., and H. Ishii. 2007. Mechanical constraints as computational constraints in tabletop tangible interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 809–18. New York: ACM.

Patten, J., H. Ishii, J. Hines, and G. Pangaro. 2001. Sensetable: A wireless object tracking platform for tangible user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 253–60. New York: ACM.

Patten, J., B. Recht, and H. Ishii. 2002. Audiopad: A Tag-based Interface for Musical Performance. In *New Interfacefor Musical Expression*, 1–6. Singapore: National University of Singapore.

Paulos, E., and J. Canny. 1998. PRoP: Personal roving presence. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 296–303. New York: ACM.

Perlman, R. 1976. Using computer technology to provide a creative learning environment for preschool children. In *MIT Lego Memo, #24*. MIT M.S. Thesis. http://dspace.mit.edu/handle/1721.1/5784, http://hdl.handle.net/1721.1/5784.

Piper, B., C. Ratti, and H. Ishii. 2002. Illuminating clay: A 3-D tangible interface for landscape analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing our World, Changing Ourselves*, 355–62. New York: ACM.

Poupyrev, I., T. Nashida, and M. Okabe. 2007. Actuation and tangible user interfaces: The Vaucanson duck, robots, and shape displays. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 205–12. New York: ACM.

Pousman, Z., and J. Stasko. 2006. A taxonomy of ambient information systems: Four patterns of design. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, 67–74. New York: ACM.

Raffle, H. S., A. J. Parkes, and H. Ishii. 2004. Topobo: A constructive assembly system with kinetic memory. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2004)*, 647–54. New York: ACM.

Ratti, C., Y. Wang, B. Piper, H. Ishii, and A. Biderman. 2004. PHOXEL-SPACE: An interface for exploring volumetric data with physical voxels. In *Proceedings of the 2004 Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 289–96. New York: ACM.

Rauterberg, M., M. Fjeld, H. Krueger, M. Bichsel, U. Leonhardt, and M. Meier. 1998. BUILD-IT: A planning tool for construction and design. In *CHI '98 Conference Summary on Human Factors in Computing Systems*, 177–78. New York: ACM.

Redström, J., T. Skog, and L. Hallnas. 2000. Informative art: Using amplified artworks as information displays. In *Proceedings of DARE 2000 on Designing Augmented Reality Environments*, 103–14. New York: ACM.

Reed, M. 2009. Prototyping digital clay as an active material. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, 339–42. New York: ACM.

Rekimoto, J., B. Ullmer, and H. Oba. 2001. DataTiles: A modular platform for mixed physical and graphical interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 269–76. New York: ACM.

Resnick, M., F. Martin, R. Berg, R. Borovoy, V. Colella, K. Kramer et al. 1998. Digital manipulatives: New toys to think with. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 281–87. New York: ACM.

Reznik, D. S., and J. F. Canny. 2001. C'mon part, do the local motion! In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, Vol. 2233, 2235–42. Washington, DC: IEEE.

Ryokai, K., S. Marti, and H. Ishii. 2004. I/O brush: Drawing with everyday objects as ink. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 303–10. New York: ACM.

Sankaran, R., B. Ullmer, J. Ramanujam, K. Kallakuri, S. Jandhyala, C. Toole et al. 2009. Decoupling interaction hardware design using libraries of reusable electronics. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, 331–7. New York: ACM.

Shaer, O., and E. Hornecker. 2009. Tangible user interfaces: Past, present, and future directions. *Found Trends Hum Comput Interact* 3(1–2):1–137.

Shaer, O., N. Leland, E. H. Calvillo-Gamez, and R. J. K. Jacob. 2004. The TAC paradigm: Specifying tangible user interfaces. *Personal and Ubiquitous Computing*, 8:359–69.

Singer, A., D. Hindus, L. Stifelman, and S. White. 1999. Tangible progress: Less is more in somewire audio spaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*, 104–11. New York: ACM.

Smith, D. 1982. Designing the star user interface. *Byte* 7(4):242–82.

Stifelman, L. J. 1996. Augmenting real-world objects: A paper-based audio notebook. In *Conference Companion on Human Factors in Computing Systems: Common Ground*, 199–200. New York: ACM.

Streitz, N. A., J. Geissler, T. Holmer, S. I. Konomi, C. Mueller-Tomfelde, W. Reischl et al. 1999. i-LAND: An interactive landscape for creativity and innovation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*, 120–7. New York: ACM.

Strong, R., and W. Gaver. 1996. Feather, scent and shaker: Supporting simple intimacy. In *Proc. of CSCW '96*, 29–30. New York: ACM.

Sutherland, I. E. 1964. Sketchpad: A man-machine graphical communication system. In *DAC '64: Proc. of the SHARE Design Automation Workshop*, 6.329–46. New York: ACM.

Suzuki, H., and H. Kato. 1993. AlgoBlock: A tangible programming language—a tool for collaborative learning. In *The 4th European Logo Conference*, 297–303.

Thacker, C., E. McCreight, B. Lampson, R. Sproull, and D. Boggs. 1984. Alto: A personal *computer.* In *Computer Structures: Principles and Examples*, 2nd ed., ed. Siewiorek, Bell, and Newell, 549–72. New York: McGraw-Hill.

Ullmer, B. 2002. Tangible interfaces for manipulating aggregates of digital information. New York: ACM. MIT Ph.D. Thesis. http://dspace.mit.edu/handle/1721.1/29264, http://hdl.handle.net/1721.1/29264.

Ullmer, B., Z. Dever, R. Sankaran, C. Toole, C. Freeman, B. Cassady et al. 2010. Cartouche: Conventions for tangibles bridging diverse interactive systems. In *Proc. of TEI '10*, 93–100. New York: ACM.

Ullmer, B., and H. Ishii. 1997. The metaDESK: Models and prototypes for tangible user interfaces. In *Symposium on User Interface Software and Technology (UIST '97)*, 223–32. New York: ACM.

Ullmer, B., and H. Ishii. 2000. Emerging frameworks for tangible user interfaces. *IBM Syst J* 39(3&4):915–31.

Ullmer, B., H. Ishii, and D. Glas. 1998. mediaBlocks: Physical containers, transports, and controls for online media. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, 379–86. New York: ACM.

Ullmer, B., H. Ishii, and R. J. K. Jacob. 2003. Tangible query interfaces: Physically constrained tokens for manipulating database queries. In *INTERACT 2003 Conference*, 279–86. Amsterdam: IOS Press.

Ullmer, B., H. Ishii, and R. J. K. Jacob. 2005. Token + constraint systems for tangible interaction with digital information. *ACM Transactions on Computer-Human Interaction* 12:81–118.

Ullmer, B., R. Sankaran, S. Jandhyala, B. Tregre, C. Toole, K. Kallakuri et al. 2008. Tangible menus and interaction trays: Core tangibles for common physical/digital activities. In *Proc. of TEI '08*, 209–12. New York: ACM.

Underkoffler, J., and H. Ishii. 1998. Illuminating light: An optical design tool with a luminous-tangible interface. In *Conference on Human Factors in Computing Systems (CHI '98)*, 542–49. New York: ACM.

Underkoffler, J., and H. Ishii. 1999. Urp: A luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*, 386–93. New York: ACM.

Underkoffler, J., B. Ullmer, and H. Ishii. 1999. Emancipated pixels: Real-world graphics in the luminous room. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 385–92. New York: ACM.

van den Hoven, E., and B. Eggen. 2004. Tangible computing in everyday life: Extending current frameworks for tangible user interfaces with personal objects, 230–42. Springer: Berlin.

Villar, N., and H. Gellersen. 2007. A malleable control structure for softwired user interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, 49–56. New York: ACM.

Weiser, M. 1991. The computer for the 21st century. *Sci Am* 265(3):94–104.

Wellner, P. 1993. Interacting with paper on the DigitalDesk. *Commun ACM* 36(7):87–96.

Wisneski, C., H. Ishii, A. Dahley, M. Gorbet, S. Brave, B. Ullmer et al. 1998. Ambient displays: Turning architectual space into an interface between people and digital information. In *International Workshop on Cooperative Buildings (CoBuild '98)*, 22–32. Springer: Berlin.

Zigelbaum, J., and C. Csikszentmihalyi. 2009. Reflecting on tangible user interfaces: Three issues concerning domestic technology. In *CHI'07 Workshop on Tangible User Interfaces in Context and Theory.* Scientific Commons.

Zuckerman, O., and M. Resnick. 2004. Hands-on modeling and simulation of systems. In *Proceeding of the 2004 Conference on Interaction Design and Children: Building a Community*, 157–58. New York: ACM.

Zufferey, G., P. Jermann, A. Lucchi, and P. Dillenbourg. 2009. TinkerSheets: Using paper forms to control and visualize tangible simulations. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, 377–384. New York: ACM.

22 Achieving Psychological Simplicity

Measures and Methods to Reduce Cognitive Complexity

John C. Thomas and John T. Richards

CONTENTS

22.1 INTRODUCTION: SCOPE AND STRUCTURE OF THIS CHAPTER

In this chapter, we explain why psychological complexity is (or should be) of interest to the designers of human–computer systems. We then distinguish between intrinsic complexity and undue complexity. We presume that undue complexity is generally (but not universally) counterproductive in that it leads to more errors, frustration, and greater task completion times. Generally, these are all things to be avoided in a work-oriented context. In a more aesthetic, recreational, or pleasure-oriented context, however, increased psychological complexity can often be desirable. We then examine the sources of undue complexity. We hypothesize that undue complexity can arise in the following three ways: (1) intentionally, (2) through incompetence, or (3) most commonly, as an unintended side effect of normal socialization processes.

We then differentiate complexity from many related concepts such as uncertainty, obscurity, and difficulty. We review various approaches to measuring complexity. We suggest where and how complexity may be introduced during the overall development of human–computer systems and several approaches that may help minimize undue complexity including making better use of the human capacity for performing "Theory of Mind" tasks. Finally, we speculate on some possible future developments in the field of psychological complexity and briefly discuss a number of case studies in which various approaches have been tried to reduce complexity.

22.1.1 Why Study Psychological Complexity?

Although mathematicians have treated complexity as a topic for a long time, more recently this interest has spread into many fields (see e.g., Holland [1995]; Bar-Yam [1997, 2000]). The attempts to understand fields as diverse as economics, ecology, biology, and machine learning, among others, relate to similar mathematical treatments, often under the general rubric of "Complex Adaptive Systems." Although there are still many unsolved problems in this field, one might raise the issue of why there needs to be a separate inquiry into the nature of psychological complexity. There are two basic reasons why psychological complexity deserves a separate treatment. First, what may be thought of as objectively complex, may or may not be psychologically complex. Indeed, an exploration of these differences forms a major part of this chapter. Second, when we consider complexity in the more specific context of human–computer interaction (HCI), it is useful to differentiate *intrinsic* complexity from *undue* or gratuitous complexity. Some tasks are inherently complex. We may help people perform these tasks via work aids, education, documentation, rule-based systems, or the clever design of work groups, but some considerable intrinsic complexity may remain. In contrast, although regrettable, it still seems to be a fact of life that many systems, applications, and artifacts are unnecessarily complex. Poor design, for instance, burdens users with complexity above and beyond what is required by the nature of the task. We will, in this chapter, explore sources of this "undue" complexity as well as ways to prevent or mitigate it. Although this chapter will reference some of the relevant literature on the more general topics of "complexity" and "psychological complexity," the focus of this chapter is on how we can use these concepts to improve HCI, in most cases by reducing undue complexity.

In terms of the underlying cause, there seem to be three main reasons for undue complexity. First, undue complexity sneaks into systems even though people consciously try to prevent it. This can happen for numerous reasons which we will explore in greater detail below, for example, lack of appropriate methodology. Second, seemingly undue complexity can be injected into systems *intentionally*. For instance, systems commonly enforce rules governing both password structure and password lifetime. These are often perceived as an annoyance by end users but are intentional policy decisions aimed at improving overall system security. A third and more subtle reason for undue complexity relates to ordinary social processes. As a group of people work together or live together, they develop and evolve ways of referring to things that become convenient short hands for the "in-group" but become increasingly obscure or difficult for the "out-group." In the extreme, this results in different accents and dialects and eventually distinctly different natural languages, customs, and assumptions. Disparate groups of people end up, not simply using different terms, but thinking about the world differently (Abley 2005). This process is typically, though not exclusively, unconscious. This type of socialization begins early; experiments indicate that children as young as several weeks old are already *less capable* than they were at birth of making auditory distinctions within the phonemic categories deemed equivalent by their linguistic community. In one experiment, as children living in an English-speaking environment aged from 6 to 12 months, their percentage of discriminations of a non-English distinction decreased from 80% correct to 10% (Werker and

Tees 1984). Categorical speech perception is only one fairly dramatic form of a process that is happening all the time and at many levels in normal social interaction.

22.1.2 Nature of Psychological Complexity as a Variable

Why are people interested in psychological complexity? Basically, the intuition is that it is an important intervening variable that simplifies analysis and prediction. On the one hand, there are a number of factors, explored in detail in this chapter, that impact psychological complexity and on the other hand, there are a number of dependent variables which psychological complexity influences.

One might yet question the utility of "complexity" as a unifying variable. Although Card, Moran, and Newell (1983) talk about "complex behavior" (and the principle of "Rationality" to help explain how to build models of complex behaviors from simple ones), they do not treat "complexity" as a variable, per se. Despite this, under a wide variety of conditions, accurate predictions of complex human behavior are possible (see also, e.g., John [1990] and Gray et al. [1990]). Modeling at this level of detail requires a substantial amount of work although CogTool (Harris, John, and Brezin 2010), which will be discussed in more detail in Section 22.7.5, greatly reduces this workload. The hope for "complexity" is that one might develop predictive measures of complexity that could be applied fairly easily without special expertise that would still be highly predictive of human behavior. In the ideal case, for instance, a development team might calculate the complexity of several alternative design ideas for a human–computer system and make at least some of the many necessary design decisions without the need for assessing each one with costly behavioral observations.

We expect intuitively that increased psychological complexity is positively correlated with increased errors, increased time to complete a task, and decreased productivity, as well as increased frustration in a "results-oriented" work environment. (Even in such work contexts, there may be exceptions; e.g., when lengthy vigilance at an overly simple task may actually produce more errors than a somewhat more complex one). Nonetheless, the overall presumption, both in the public mind and in the scientific literature is that unnecessary complexity is to be generally avoided on the grounds of reducing errors and improving productivity. The cost of complexity can be considerable. For example, according to a recent study by den Ouden (2006), half of the consumer products returned as "malfunctioning" are actually functioning as designed; they are just too complex for the customer to use. A well-meaning attempt to "improve" a homeland security system resulted in more errors, longer decision times, and a decreased probability that people would even use the system (Coskun and Grabowski 2005). Although details vary, Figure 22.1 shows the general relationship of complexity to performance and frustration.

However, the relationship of psychological complexity to more ludic (pleasure-oriented) and a aesthetic variables is more complex. In general, the prevailing wisdom is that a moderate amount of complexity is esthetically pleasing (see Figure 22.2). Stimuli, responses, mappings between stimuli and responses, or contexts that are *either* too complex *or* too simple are not as interesting, pleasurable, or engaging as those that are moderately so. This seems to be a fairly robust phenomenon applying to infants as well as adult humans and to various species. Of course, it is still the case that what constitutes moderate complexity for an individual depends on previous knowledge as well as personality predilections for more or less complexity. In any case, psychological complexity is potentially a powerful intervening variable that, on the one hand, promises to collapse the impact of numerous separate independent variables into one (i.e. "complexity") and, on the other hand, to expand the implications of complexity onto a number of dependent variables. In the case of performance-related variables, the relationship is thought to be monotonic; in the case of subjective variables, the relationship is thought generally to be an inverted U-function although it may vary according to the sense and the individual.

In the last few decades, most HCI work has examined work contexts in which productivity, in the broadest sense, has been a primary focus. As computing technology has become more intertwined with home life and entertainment, the importance of also using psychological complexity as a predictor of pleasure and preference is increasing. This implies that HCI practitioners must take care not to presume that *minimal* complexity is necessarily the *best level* of complexity.

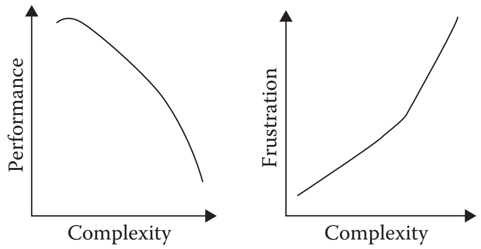


FIGURE 22.1 The general relationship between complexity and performance and complexity and frustration.

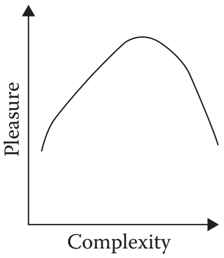


FIGURE 22.2 The typical relationship between complexity of a stimulus and the associated aesthetic pleasure in ludic situations.

In the domains of art and architecture, some attempts have been made to quantify what makes for "interesting" or "aesthetically pleasing" patterns. For example, Christopher Alexander's work on *The Nature of Order* (2002) includes an examination of and ordering of visual binary patterns, and Klinger and Salingaros (2000) suggest a metric for measuring the visual complexity of simple arrays. Kidd, Piantadosi, and Aslin (2010) show that infants prefer moderate complexity. Other recent approaches attempt to relate preferences to fractal numbers (Taylor et al. 2005). Other investigators have tried to predict musical complexity based on objective factors (see e.g., Pressing [1999]; Shmulevich and Povel [2000]; Shmulevich et al. [2001]).

In more work-oriented contexts, psychological complexity is clearly related to other intervening variables such as stress (Figure 22.3), difficulty (Figure 22.4), and workload (Figure 22.5). However, there are distinctions among these. Stress depends not just on the complexity of the problem situation itself but also on external factors such as the payoff matrix and internal factors such as neuroticism. For instance, a specific puzzle might be thought to have a certain level of complexity. However, the amount of stress that one experiences in attempting to solve it would depend on the perceived rewards, punishments, and time pressure as well as the individual's internalized habits for viewing how various outcomes relate to overall life goals; for example, proclivities for "awfulizing" (Ellis 2001) tend to increase stress far more than what might be "objectively" justified.

The more complex the task, the more difficult we would expect it to be; other things being equal. However, difficulty can also accrue to a task for a number of other reasons including environmental stressors such as extreme heat or cold, vibration, noise, or the necessity of applying large forces.

Workload is a term more commonly applied to an individual or a team *across* tasks. So, one can imagine an individual with a large number of tasks, each of which is fairly simple, but whose job includes many interruptions and context switches from one "simple" task to another. In such a case, the workload may be fairly high even though any particular task may be fairly simple. It is sometimes useful to make a similar distinction within a task. Programming, for instance, often involves some mix of creative tasks that deal with the essential problem and a number of administrative overhead tasks that must be accomplished as well. In our work on reducing complexity for high-performance computing, we are attempting to quantify the impact that reducing the administrative overhead may have, not only in directly reducing overall time, but also in terms of allowing the programmer to concentrate more effectively on the creative aspects of the task.

There is another sense in which the term "psychological complexity" is sometimes used. Social critics may use the term to refer not to a single task or system but to the totality of life experiences. Modern life in the information age is sometimes deemed to be too complex *in its totality.* This increased complexity may apply to politics, personal relationships, child-rearing, transportation, health care, food choices, and so on extending potentially to every aspect of modern life. Even providing for our entertainment needs has become a complex endeavor (Grinter and Edwards 2005). In addition to the notion that each separate aspect of life is becoming more complex, people often find themselves multitasking and task switching (e.g., Gonzalez and Mark 2005). Although this is a potentially interesting avenue to explore in its own right, it is basically beyond the scope of this chapter, which instead will focus on psychological complexity as it applies to individual systems or tasks that have information technology aspects.

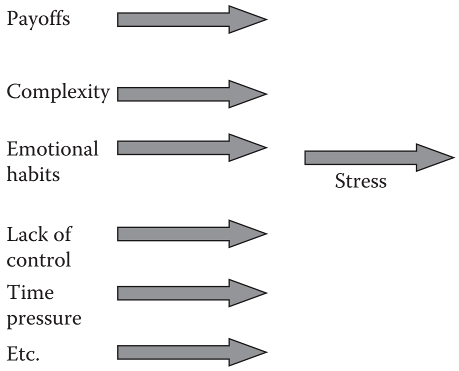


FIGURE 22.3 A number of variables, including complexity, can increase stress.

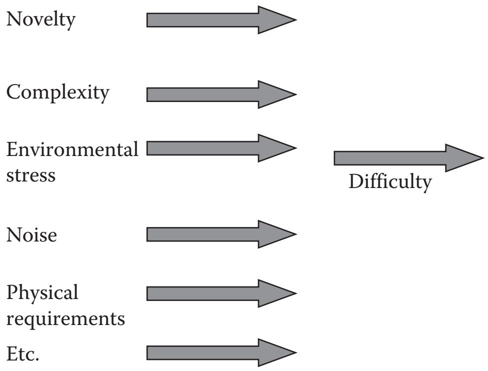


FIGURE 22.4 A number of variables, including complexity, can add to task difficulty.

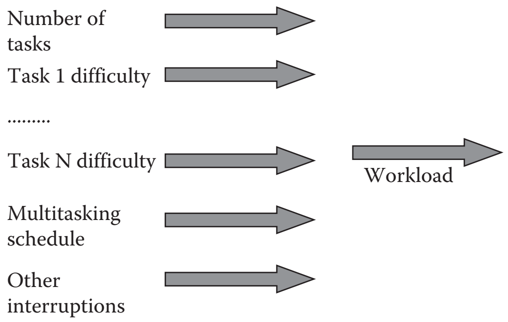


FIGURE 22.5 Overall workload is influenced by the difficulty of various tasks, the number of tasks, the scheduling of the tasks, and other factors.

Early attempts to measure task difficulty with objective metrics include Attneave's (1957) figure complexity. There have been attempts to provide measures for the "meaningfulness" of trigram "nonsense" stimuli, responses, and the mapping between the stimulus and response in the verbal learning literature. Others have attempted to extend this work to real words; English words have been rated for frequency of occurrence (Kucera and Francis 1967; Brown 1984), concreteness (Paivio, Yuille, and Madigan 1968), and so on. In one case, for example, (Rubin 1980), *fifty-one* dimensions of words were measured. Such measures do provide predictability in a wide range of laboratory tasks. For instance, the log of word frequency is closely related to decreased latency in a simple naming task where subjects are shown pictures and asked to name the pictured object as quickly as possible (Thomas, Waugh, and Fozard 1977). However, such empirical relationships as these are problematic in terms of applicability to HCI. For one thing, when measures are taken on real words in English, many dimensions tend to be correlated. Work by Carroll and White (1973), using a linear regression model, showed that *age of acquisition* is actually a more powerful predictor than frequency of occurrence (though the two are highly correlated). Neural net models of learning also demonstrate this effect (Smith, Cottrell, and Anderson 2001). How might one apply such findings to HCI? For example, given a choice between two otherwise equally appropriate words, it is generally better to use the word that is more frequent in the language, or even better, one that is learned early in life. The relevance of this heuristic to the design of instructions, web pages, error messages, and so on, however, is very limited because in real design situations, we seldom have a choice between two "otherwise equally appropriate words." More frequent words also often have the property of being more ambiguous, both semantically and syntactically (e.g., "can," "run") than longer, less frequent words. Words learned early in life often have little relevance to most tasks that adult users engage in. Generally, knowledge of the particular users, their vocabulary, task demands, conventions, contexts, and so on dominates more general properties of the language in suggesting the best wording.

22.2 COMPLEXITY AND RELATED CONCEPTS

In this section, we attempt to further define psychological complexity by distinguishing it from a number of similar concepts.

22.2.1 Relationship of Complexity and Ease of Use

Intuitively, it would seem that "ease of use" and "simplicity" constitute closely related concepts. However, they do differ in several ways. First, "ease of use" implies an empirical orientation. There are a number of interesting issues involved in measuring "ease of use" such as choosing the set(s) of reference users, representative tasks, and in choosing the dependent variables to be measured (time to complete, errors, quality of result, etc.). "Ease of use" can, in principle, be measured objectively in terms of human behavior. Complexity, on the other hand, does not have the same degree of conceptual consensus. There are approaches that focus on the formal, intrinsic properties of a system or stimulus (e.g., Halstead's *Software Physics* [1977]; Christopher Alexander's recent work on *The Nature of Order* [2002]). Other approaches focus on the reactions of a human being to the system or stimulus. These can include passive measures such as pupil dilation, staring time, or heart rate deceleration. They can also include more subjective judgments or attempts to measure various aspects of task-oriented behavior, for example, the number of steps in a process or the number of variables that must be remembered (e.g., Brown and Hellerstein 2004; Brown, Keller, and Hellerstein 2005).

Nonetheless, other things being equal, one would expect something that is less complex to be easier to use. However, other things are typically *not* equal. "Ease of use" often depends heavily on people's expectations that may result from cultural conventions, often varying widely with time and place. Today, a new application that uses the typical graphical user interface (GUI) widgets such as pull-down menus may be fairly easy to use for experienced users, primarily because people have a fair amount of positive transfer from earlier, similar experiences. A unique and unfamiliar but "less complex" interface may not prove as easy to use.

In addition, a design that maximizes visual simplicity through symmetry and minimalism may actually increase the difficulty of use. For example, a shower arrangement with one, clearly marked handle for hot water and another for cold water, combined with a two-position lever for shower and bath is more complex visually than a single, unlabeled knob with several degrees of freedom. At least initially, however, the former will be easier to use. Similarly, a "normal" bottle opener is not particularly elegant or simple in shape. However, its asymmetries, as well as its commonality, make it easy for people (in our culture) to use. An alternative bottle opener (see Figure 22.6) consists of a metal cone with a thin protruding knob. This device is physically "simpler," but harder to use, at least initially.

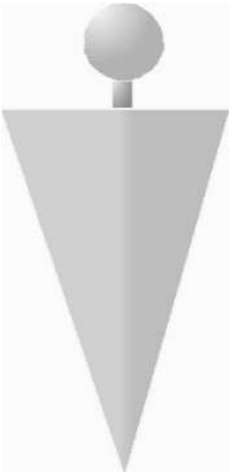


FIGURE 22.6 A "simple" but highly nonintuitive bottle opener.

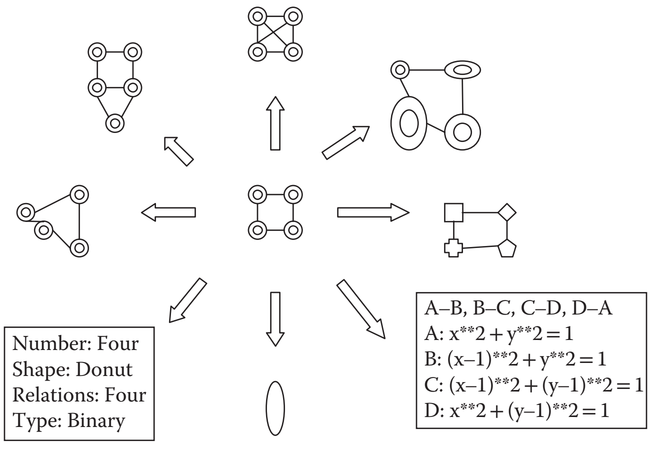
22.2.2 Relationship of Simplicity and Complexity

Intuitively, simplicity and complexity seem to be opposites. Indeed, they are listed as opposites in Roget's Thesaurus (2005). However, there is an interesting asymmetry hidden in their opposition. It is as though absolute "simplicity" is the center point of an *n*-dimensional sphere while, absolute "complexity" is the surface. In other words, there are many "ways" of moving away from simplicity toward complexity. Figure 22.7 illustrates just a few of the many ways that a system can become more complicated. For simplicity, we illustrate a two-dimensional rather than *n*-dimensional space.

This is more than an idle philosophical observation. In the real-world iterative cycle of design and behavioral observation, it is relatively difficult to move "inward" toward greater simplicity while improving ease of use. It is relatively easy to move along the circumference of a notional circle, resolving some types of complexity while simultaneously introducing others and therefore not increasing either overall simplicity or ease of use. For example, in designing a word processing system, you might find that people are having difficulty finding desired items in long pull-down menus. You decide to make each menu "simpler" by changing the pull-down options depending on whether one clicks or clicks with the shift key depressed. This makes each menu simpler but introduces another kind of complexity. Much of the "art" and the importance of experience in HCI can be conceived of as being able to move "in" from the surface rather than "around" the surface. See the work by Lewis (2007) for a related discussion with respect to simplicity and accessibility.

22.2.3 Relationship of Complexity and Uncertainty

Increased complexity is associated with increased uncertainty, *ceteris paribus.* Complexity is a term that can reasonably be applied to stimuli, responses, or to the mapping between them. The concept of uncertainty, however, seems to apply only to mapping. One can have uncertainty about what a stimulus is, or uncertainty about what response to make, but fundamentally, it has to do with what action is appropriate, given the current circumstances.



Number: Four
Shape: Donut
Relations: Four
Type: Binary

A–B, B–C, C–D, D–A
A: x**2 + y**2 = 1
B: (x–1)**2 + y**2 = 1
C: (x–1)**2 + (y–1)**2 = 1
D: x**2 + (y–1)**2 = 1

FIGURE 22.7 A "simple" system can be complicated in many different ways. (Here are just a few.)

Complexity, as mentioned earlier, can be thought of either as something objectively measurable or as something subjective. The term "uncertainty" also has an objective meaning. If one turns over a randomly chosen card from an ordinary deck, there is more objective uncertainty about the outcome than if one flips a coin. The term uncertainty may also be applied in a purely subjective way. In this subjective sense, a given person may be "completely certain" that a randomly chosen card will be the Ace of Spades (because of nothing more than a compelling intuition) and completely uncertain about the outcome of a coin toss. The concept of subjective uncertainty has even further shades of application. One may watch the movie *Apollo 13* on numerous occasions. At one level, the outcome is already predefined and completely known. There is neither objective nor subjective uncertainty. Yet, within the inner context of watching the story play out, the viewers "allow themselves" (each time!) to feel the subjective uncertainty of outcome.

Subjective uncertainty in HCI can apply at numerous levels. For example, to make touch typing a felicitous interaction, one subjectively presumes that each keystroke will be correctly transmitted to the computer and that what appears on the screen is an accurate mapping of what is actually being stored inside the computer. Objective certainty about the states of electromechanical systems probably never reaches 100%. However, as a *strategy* for partitioning effort, it is often useful to ignore very small error probabilities in subjective judgments. The first time that one of the authors (JT) typed a sentence with an initial "the" and had it automatically transformed into a "The" by the word processing software, this "certainty" changed. The author began to wonder "what else" the application might be doing. Later, a much worse problem surfaced in the form of an automatic update to styles. In response to something typed on page 1, something might change appearance on page 100. Subjective uncertainty also bears some resemblance to the concept of "trust." One might have 1000 interactions with someone, each of which provides evidence of "trustworthiness," yet even a *single* interaction that shows a person to be untrustworthy may color a relationship for years to come. Similarly, if an interaction between user and the computer system violates an assumed behavioral norm by doing something completely unexpected, the level of subjective uncertainty created may be much greater than warranted in any "objective sense." Moreover, if the user's experience includes such a violation, the *scope* of subjective uncertainty may be difficult to predict. One user may simply generalize uncertainty to a specific function; another may come to "distrust" a specific application (have a high degree of uncertainty), whereas a third may come to "distrust" computers, *in general.*

22.2.4 Relationship of Complexity to Number

One would expect that if systems A and B are equivalent in other respects, but system A is more extensive than system B in terms of the number of items, that system A is more complex than system B. However, the degree to which an increase in number produces an increase in complexity depends greatly on the structure of the systems. Suppose that systems A and B consist of 5 and 10 buttons, respectively, combined with an "OK" light. The user has to use trial and error to discover the correct button to push to cause the OK light to light. On average, it will take 2.5 trials to press the right button for system A and 5 trials to press the right button for system B. Here, doubling the number of choices doubles the average number of trials. On the other hand, imagine that the user must find a sequence of 5 button pushes for system A and 10 button pushes for system B. When the user hits each correct button, the OK light comes on. On average, it will take the user $2.5 + 2 + 1.5 + 1 + 1 = 8$ attempts to execute the right sequence for system A and $4.\ 5 + 4 + 3.5 + 3 + 2.\ 5 + 2 + 1.5 + 1 + 1 = 22$ attempts to find the right sequence for system B. Here, a doubling in the number of buttons results in nearly three times the mean number of trials. For a final case, imagine that the OK light comes on only after *all* buttons are pushed in the correct sequence. There are 5! possible sequences for system A, and the user will find the correct sequence on average in 5!/2 trials or 60 trials. There are 10! possible sequences for system B, and the user will find the correct sequence, on average in 1,814,400 trials. Here, a doubling in number has led to an increase in trials by a factor of 30,240. In practical terms, system A, though cumbersome, could be solved by hand in a few minutes. System B would probably extend beyond anyone's patience. A quantitative difference in complexity actually would result in the qualitative difference between something doable and something not doable. These examples illustrate that the impact of number on complexity greatly depends on the nature of the system. In particular, "hidden states" are almost always a "bad thing" in terms of complexity, but just how bad they become rises quickly with increasing number.

Another example has to do with working memory and relates to George Miller's (1956) famous paper, "The magic number 7 plus or minus 2." Although the "size" of working memory varies somewhat depending on the number of dimensions and the type of material, it is not large. A system that allows a person to operate within that span can be nearly error free, while a system that forces a person to operate beyond that span will be extremely error prone. A fairly modest increase in memory load can thus result in a very large increase in time and errors (see Figure 22.8). This is one reason we feel, for instance, that a modest increase in support for the administrative overhead of programming may result in a larger increase in overall productivity.

22.2.5 Relationship of Complexity to Nonlinearity

One of the defining characteristics of complex adaptive systems is that they exhibit nonlinearity. From the perspective of survival, this nonlinearity is a good thing. From the perspective of someone trying to interact with a computer system, nonlinearity can be a major source of difficulty in understanding, predicting, and controlling the behavior of the system. For example, one of us (JT) once had to debug a PDP-8 program that was intended to collect reaction time data. There was only a single bit wrong in the entire program (which probably had roughly 5000 bits total). In a linear system, you would expect that a 0.02% error would result in a small behavioral problem; for example, the reaction time would be off by a constant 0.02%. Of course, nothing of the kind happened. The program did not collect reaction times at all. Instead, it eventually caused the interpreter to crash. The "bit" that was wrongly set was the "indirect bit" so that instead of storing the reaction time, the program used the reaction time as an address to store the contents of a register. Furthermore, since every time the author tried to test the program, the reaction time varied slightly, the precise behavioral path of the program was different every single time. This example is not at all unusual but illustrates the property that computer systems are typically highly nonlinear in their behavior. Generally, higher layers of software are designed to make the system, as seen by the user, as being more nearly linear (and therefore, more predictable). However, even in commonplace high-level applications, many nonlinear aspects typically remain.

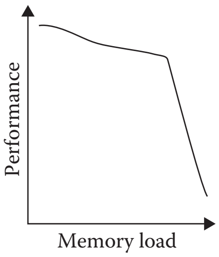


FIGURE 22.8 How memory load can have a minor effect on performance, up to a point, after which increasing memory load still more yields a sharp drop in performance.

22.2.6 Relationship of Complexity to Distribution

Halstead (1977) claimed that complexity C is equal to a constant k times the sum of the number of unique operators U times the natural log of the total number of operators u plus the number of unique operands O multiplied by the natural log of the total number of operands o.

$$C = k(U + \text{In}\, u + O + \text{In}\, o) \tag{22.1}$$

This is not an unreasonable approximation. We have already seen how number can increase complexity. However, we could also consider how these operators (or operands) distributed throughout a program. We speculate that

1. AAAAAAAABBBBBBBCCCCCCCC

is easier to comprehend, use, edit, and so on than

2. AABBCAABCBBCCAABCAACCCB

It has been known for some time that repeated operations are typically faster than alternating ones (e.g., Fozard, Thomas, and Waugh 1976). There can be exceptions to this general rule based on rhythm, fatigue, or satiation, especially for extremely simple tasks. However, for complex tasks, we generally expect a distribution such as 1 above to be much easier than 2. For example, in the Stroop task, one can be asked either to name the colors of ink (while ignoring the word) or to read the names of colors (while ignoring the color of ink). Alternating between the two every line is much slower than doing either task *en masse* (e.g., Philips et al. 2002). In the Brown, Keller, and Hellerstein (2005) model, this source of complexity is modeled as context shifts.

22.2.7 Relationship of Complexity to Nature of Elements

Given the same number and structure of elements, two systems that are formally isomorphic may be quite different in terms of psychological complexity based on differences in human biology, learning, or both. For example, it seems clear that the human visual system is especially well tuned to handle the complexity of human faces. For example, Johnson and Morton (1991) present evidence that babies as young as 3 hours after birth can recognize human faces. Chernoff (1973) suggests that this ability may be used to represent underlying complexity in a way that is easier for humans to deal with. In the auditory domain, it is clear that the difficulty of dealing with (recalling, rearranging, etc.) a string of phonemes that follow the phonological rules of one's native language is much less than dealing with an equal length string that violates those rules.

22.2.8 Relationship of Complexity to Naming Scheme

Consider two programs that manifest identical structures. The program asks the user to input an integer from 1 to 100. For each integer, the program then displays a corresponding unique horoscope. Structurally, this is a very simple program. However, imagine that in one version, the user's input number is called, "User's Input Number" and in another it is called "NXBNM." Further, suppose the horoscopes in the first program are labeled, "Horoscope1, Horoscope2, Horoscope3, etc." for each corresponding integer. In the second program, they are each labeled with a random two-letter string such as "HA," "IB," "JK," and so on. It is hopefully clear that these two programs, though formally identical would be vastly different in how easy they would be to understand, modify, and debug.

Although the above example is somewhat contrived, developing a consistent and intuitive naming scheme is a real world problem. Often, when a programmer (or any other kind of designer) begins to name things, they do not have a complete understanding of the space of things they will have to name. For instance, when JT began working on a recent project, he named a file folder "ELFL" for "Electronic Learning Flow Language." As the project grew, one folder was much too generic for all the nuances that emerged. Furthermore, the focus of the project changed and noone even talked about "Electronic Learning Flow Language" anymore. Furnas et al. (1984) found that, not only are two different people likely to come up with different names for things; even the same person over time is unlikely to spontaneously come up with the same name. They suggest tables of synonyms to help alleviate this problem. Real computer systems are often rife with names that are difficult to comprehend or to recall precisely. The use of common search engines by millions of people helps solve this problem.

22.2.9 Relationship of Complexity to Obscurity

Consider the following two sequences of binary digits:

A: 0000 0001 0010 0011 0100 0101 0110 0111 1000
B: 1000 0101 0100 0001 0111 0110 0011 0010 0000

These two sequences each contain the same number of binary digits and the same number of groupings. The first sequence is essentially a counting sequence from zero to eight. This is a fairly obvious sequence *if* one knows how to count in binary. The second sequence would presumably be much harder to memorize for most people. However, it also represents the numbers zero to eight; however, they are arranged in alphabetical order according to the English names. Once one realizes this rule, memorizing or reconstructing the second sequence becomes much easier. There is a sense in which the first sequence strikes us as a natural ordering because it is based on what we can see right before us. Numbers are likely to be highly associated with counting, after all. The rule that orders the second sequence strikes us as more obscure however since it depends on some other representation (the English names of the numbers) that is not naturally and strongly associated with the numbers qua numbers. This notion of "obscurity" is closely associated with the contrary concept of "affordance" (Norman 1988). Basically, if the perceptible properties of an object or system immediately make it clear what it can do and how to do it, we can say that the object or system has good affordances. Of course, whether something appears obscure or not depends heavily on the user's previous experience, both cultural and personal, and may include inborn factors as well. In Figure 22.7, the "equations" version of the simple diagram in the middle illustrates obscurity.

22.2.10 Relationship of Complexity to Structural Framework

One of the interesting aspects of perceived complexity is that when it comes to human beings, sometimes "more" is "less." For example, most English readers are capable of distinguishing "WORD" from "CORD" at a briefer presentation time than they can distinguish "W" from "C." (Reicher 1969; Hildebrandt et al. 1995). Learning a set of paired associates, presented along with mnemonic visual images, is much easier than learning the set of paired associates alone (Thomas and Ruben 1973). Subjects presented with a set of sentences and a context-setting theme or picture have a much easier time learning and remembering a story than those who do not have the picture (Bransford and Johnson 1973). Recalling a story that fits with our cultural patterns is much easier than recalling one which does not (Bartlett 1932). In all the above cases, it appears that the advantage of the structural framework is because it relates new items to something that already exists in memory.

In other cases, however, the advantage of structure seems to have more to do with the underlying nature of the nervous system. For example, in attempts to develop useful and novel representations of speech signals to improve synthesis, one technique developed by Pickover (1985) mapped an autocorrelation function into polar coordinates scaled to fit within a 30° angle. The resulting dot patterns proved singularly difficult to interpret in any useful way. However, when the pattern was reflected to include a mirror image and the resulting 60° angle repeated six times around a central point to produce a snowflake-like pattern (see Figure 22.9), the representation easily differentiated various vowel sounds as well as some subtleties like anticipatory nasalization, which were virtually invisible in the traditional sonogram. Of interest in this context is the fact that the useful representation contained no more information than did the useless one. Arguably, one could say the "speech-flake" pattern was more complex. However, the "complexity" apparently allowed visual symmetry detectors to come into play and produced far more distinguishable and memorable patterns. These symmetrized dot patterns have since been applied to numerous other domains as diverse as cardiac signals, mechanical stress, and even the crunchiness of cereals!

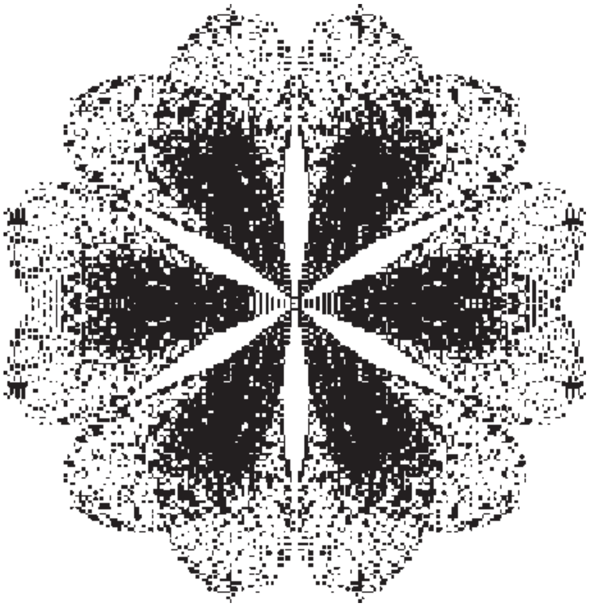


FIGURE 22.9 A symmetrized dot pattern.

22.2.11 Complexity of System versus Task Complexity

It is easy to fall into the trap of conflating the complexity of an interactive computer system with the complexity of the task that one is attempting to perform while using that system. In fact, many studies in the HCI literature compare two or more versions of systems by having subjects perform a small number of tasks in the two systems. Often these tasks are treated as fixed factors when they are actually a small sample (probably not randomly chosen) of all possible tasks. In principle, it is very difficult to generalize about the superiority of one system over another on the basis of performance on a few specific tasks. One cannot really know that a different set of tasks may not have completely obliterated or even reversed any observed trend.

As a general heuristic, one might suppose that the complexity of the system should reflect the complexity of the necessary tasks to be supported. That is, if the user only needs to perform very simple tasks, it may be enough to provide a simple system. However, if the user needs to perform very complex tasks, then a more complex tool may be required. In general, there may be some truth to this heuristic, but it must be applied with care. One caveat is that the function associated with this increased system complexity must actually be useful and allow the user to focus attention on the task at hand. Otherwise, additional complexity in a system may actually be *more disruptive* when the task is complex than when it is simple. For example, consider the tasks of writing a two-page trip report and writing a full-length novel. Writing the novel (we assume) is a much more complex task. It may be that the various functions, fonts, formatting, and options in a complex word processor are unnecessary and even more distracting for someone trying to keep in mind a complex set of subplots and characters than for someone writing a short trip report.

Another important distinction is between the underlying complexity of a system and the complexity that is surfaced to the user. A huge amount of research and technological sophistication may be involved in the development of an automatic transmission but from the driver's perspective, the transmission is simple. From the perspective of the user, modern search engines provide a simple, commonplace interface. Most end users have no notion of the complexity of the underlying processes. In fact, even *developers* apparently do not always see the necessary complexity behind providing common user interface functionality such as "undo." Apparently, the provision of architectural patterns can aid in ensuring the necessary insights (John et al. 2009).

Probably, the ideal case is to provide a layered interface in which the system reveals only the complexity needed for the task at hand. Early examples of this approach include the concept of "training wheels" (Carroll and Carrithers 1984). The interfaces for another related set of projects, the Speech Filing System, the Audio Distribution System, and the Olympic Message System (Gould and Boies 1984), also provided layered interfaces. However, in many real systems, it often happens that the user inadvertently "falls through" the user interface into a deeper layer or *must* do so to accomplish their real task. In addition, undue complexity in the underlying system can *indirectly* hurt the user experience by making code harder to test, debug, document, and maintain. Therefore, we consider undue complexity as something to be generally avoided both internally and in terms of what is to be made visible to the end user.

22.2.12 Complexity of System versus Contextual Complexity

A person may be using a system, which is fairly simple; for example, a bicycle, and their task may be fairly simple: riding in a fairly straight line without falling or crashing. However, riding that bicycle in a straight line on a dedicated and fairly empty bike path through a park might be quite different from riding on a busy city street. It is somewhat of a judgment call to draw a line between "task" and "context." In this example, one might just as well say that the two tasks are different. However, collapsing all such variations into the task complexity is probably counterproductive because the types of actions that can be taken and the power to take those actions are typically quite different in the case of simplifying the task versus simplifying the context.

Contextual complexity could increase overall complexity by providing secondary tasks, or by bombarding the person with extraneous stimuli. Contextual complexity might also produce internal distracting material. Imagine two students using a word processor to complete a take-home essay exam. One of them is also worried about the outcome of some diagnostic medical tests. Or, consider two executives who are using presentation software to construct a sales pitch. But one of them is operating in an organizational context that makes it necessary to pass the presentation through four layers of management, each with very different ideas about what makes a good sales presentation. It seems much more natural to consider such differences to be differences in contextual complexity rather than task complexity. We might expect the design of the presentation software to help the user deal with the task complexity in this example, but not the contextual complexity.

22.2.13 Complexity, Feedback, and Interactivity

Finally, the extent to which we can easily understand, control, and predict the behavior of a system has much to do, not only with how complex it is but how readily we can interact with it and receive timely and unambiguous feedback. For example, the "dynamic query" system (Ahlberg, Williamson, and Shneiderman 1992) does not decrease the complexity of the underlying data that one is attempting to understand, nor does it decrease the complexity of the interface by which one browses that data. In fact, the interface is actually more complex than a static browsing interface might be. However, what it *does* allow is fast, continuous, unambiguous feedback. Conversely, it is well known that delayed auditory feedback (or delayed visual feedback possible with television circuits) is highly disruptive to performance. In some settings, feedback on performance can still present complexity in the form of credit assignment. If you lose a game of chess, it is unambiguous that you lost, but determining what caused you to lose is difficult. This is not just a "human" problem; the "credit assignment" issue is crucial to any complex adaptive system (Holland 1995). In golf, putting can be a very difficult skill to improve. The main reason is that it is difficult to determine which of many possible factors is responsible for success or failure in putting in a real game situation. For example, if you miss a putt to the left, it might mean that you misread the slope of the green; misread the grain of the green; hit the ball off center of the putter; hit the ball with a curved arc swing; hit the ball with a blade not normal to the path and all combinations of these factors. In the domain of HCI (and elsewhere), this lack of unambiguous feedback about which subset of actions is causing which difficulties has sometimes been referred to as "tangled problems" (e.g., Carroll and Mack [1984]). In the domain of golf, Dave Pelz, an MIT physicist and ex-astronaut, has invented a series of devices to give the learner unambiguous and differentiated feedback about these various possible sources of error (Pelz 2000). Studying these devices might be a valuable exercise for system designers hoping to provide a similar untangling for users.

22.3 FOUR SOURCES OF DIFFICULTY FOR HUMAN–COMPUTER INTERACTION

One goal related to psychological complexity is to provide a system that will allow nonprogrammers to do what is essentially programming; that is, allow them to get a computer system to do what they want it to, not just in terms of choosing from preexisting options but to allow them to have more open-ended productive control. A whole succession of projects at Carnegie Mellon University (see, e.g., Myers, McDaniel, and Kosbie [1993]), the LOGO and related projects at MIT (e.g., Papert 1993); most recently, Scratch (Resnick et al. 2009) and the long series of projects by Alan Kay and others leading most recently to Squeak and Etoys have attempted to reach this goal. Much of the focus of these projects was based on trying either to simplify the syntax and semantics of communicating with the computer or allowing it to take place in a manner more nearly like the way people communicate with each other. It is certainly true that the detailed and often obscure syntax of "ordinary" computer languages can provide a barrier to use. Weinberg (1971) pointed out, for example, that FORTRAN then had one set of rules for what constituted a valid arithmetic expression when used in an assignment statement and another, more restrictive set of rules for what constituted a valid arithmetic expression in an array index. Typically, programming languages are full of these kinds of details and no doubt, they do provide unnecessary complexity for someone who does not spend a large amount of time using such a language. However, it is important to note that such details are only one of four potential sources of difficulty people may have in communicating their desires to a real computing system. For concreteness, consider a scenario in which a chess grandmaster without programming experience desires to write a computer program that plays excellent chess.

22.3.1 Understanding the Syntax and Semantics of Communicating with the Computer

As stated, the chess grandmaster will probably have to learn the somewhat arbitrary syntax and semantics of a programming language. Although there is still promise in techniques aimed at programming by example and natural language communication, at this point, such systems are insufficient for writing something as complex as a chess program.

22.3.2 Making Tacit Knowledge Explicit

Assuming that the expert can master the rules of some programming language and its associated development environment, there are other difficulties. One of these is that the chess expert, like experts in many fields, is unlikely to be consciously aware of all the knowledge that he or she brings to bear on the game of chess. Although some of this knowledge is somewhat explicit; for example, "develop the center," "protect your king," "look for double attacks," much more of it exists in the form of patterns that have developed over the course of many experiences (DeGroot 1978; Simon and Barenfeld 1969; Simon and Gilmartin 1973). Making this tacit or implicit knowledge explicit, for fields of any depth and breadth, will probably prove a far more difficult and time-consuming task than learning the details of a specific programming language. The methods and techniques of knowledge programming and expert systems may be of some help here along with storytelling (Thomas 1999; Thomas, Erickson, and Kellogg 2001). Typically, one may elicit additional tacit knowledge (i.e., transform it from tacit to explicit) by encouraging the expert to recall specific instances, asking them to generalize, and then asking for counterexamples. It can also be useful to involve pairs or small groups in exchanging stories about their experiences.

22.3.3 Making the Computation Efficient and Effective

Another set of issues revolves around the efficiency of what is going on "under the covers." Although computers are becoming ever more powerful and cheaper, it is still easy for nonexperts to write programs that are so inefficient as to be unworkable. A chess program that theoretically makes good moves but takes years for each move is unworkable and unusable. In fact, a naive approach to writing a chess program is simply to do an exhaustive search to all finishes and work backwards. It may or may not be obvious to someone untrained in mathematics or computer science that such an approach would be completely unworkable.

Although the chess case may be extreme, far less extreme cases can still be quite problematic. If one is writing an interactive program, timing issues are important and how to achieve good timing may require a great deal of expertise and knowledge. If systems involve multiple people and or computers, it is quite easy to introduce inconsistencies, deadlocks, thrashing, and so on even if a "nonprogrammer" can master syntax and make implicit knowledge explicit. For a deeper examination of the relationship of usability and choices in the underlying architecture, see the work by Bass and John (2003).

22.3.4 Making the System Understandable and Maintainable

In the ongoing stream of behavior, we make what seem to be "obvious" choices, such as where to put the car keys. If, a day later, we are prone to forget this "obvious" choice, imagine how much more difficult it can be to understand and recall the numerous decisions that must be made in designing and implementing a computer system. A nonprofessional programmer may have very little knowledge of documentation, help systems, updates, security, compatibility issues, the process for reporting and fixing bugs, and so on. As a consequence, even if the first three hurdles are overcome and a workable program is produced, its life may be very short indeed.

To summarize, the goal of having nonprogrammers directly instruct computers in a generative way has often focused on designing a very simple and consistent syntax for a programming language. In Sections 22.3.1 through 22.3.4, we indicated that such a focus only addresses the first of four major stumbling blocks end users face: to wit, learning a complex syntax.

Although the possibility of programming by nonprogrammers is intriguing and probably quite a bit more difficult than it at first appears, there is a slight variant that is becoming more commonplace. There are instances where a community of practice includes some individuals who, although perhaps not professional programmers, are quite proficient at finding and modifying code to fulfill certain functions. In some cases, they may be using programming-like functions in an application program such as a spreadsheet. In other cases, they may actually be using full-fledged programming languages. By modifying existing code incrementally, these semi-expert users are often able to address all four of the issues mentioned above. An interesting early example of such a community was "Moose Crossing" (Bruckman 1997), which provided a shared environment for kids to teach each other object-oriented programming in a special simplified language. Nonetheless, there remain many applications and systems that users interact with that are developed by professional programmers working in software companies. An examination of the way in which products are developed, in turn, can provide some insight into various places where undue complexity may be injected into systems (and therefore, how they might be prevented).

22.3.5 Conceptualizing Human–Computer Interaction as a "Theory of Mind" Task

To perform correctly on the so called Theory of Mind task (Premack and Woodruff 1978) requires that the subject be able to "put themselves in the place" of another. In a prototypical task, a treat might be hidden under a pillow in view of the subject A and another person B. The second person B leaves the room and, in full view of A but not B, the treat is moved to another location, say, on a shelf. Now you ask A, "When B returns, where will they look for the treat?" A very young child will say "on the shelf" because he or she knows that that is where the treat is. Most children over five will say "under the pillow" because they understand that it is behavior consistent with the state of knowledge of person B. Adults with autism spectrum disorders may have considerable difficulty with this type of task. However, most adults seem to have the *capacity* to put themselves in another's shoes, at least when it comes to physical knowledge. It is more problematic when it comes to understanding that someone may behave differently because of differences in culture or language. For example, one of us (JT) working on a multinational team suggested that one of the team members might want to record the pronunciation of their name in a corporate directory for the benefit of other team members. He replied, "Oh, there's no need because my name is quite simple—Johnston." He did not seem to realize that such a name might *not* be simple for someone whose native tongue was not English. In a workshop on cross-cultural issues in HCI at CHI 92, we used a card game in which people at different tables were initially given slightly different sets of rules. All went well until people began switching tables, at which point different sets of rules came into conflict. What is interesting about this exercise is that people initially attributed other people's behavior to incompetence or greed and only much later did the possibility that they were operating under different sets of rules surface. This was in a workshop whose explicit theme was cross-cultural issues! In a similar manner, one can imagine that it may be difficult for developers to imagine that end users would not have the same kinds of concerns and goals that they do.

So here we have an interesting paradox. On the one hand, the ability of people to "put themselves in another's shoes" greatly simplifies what would otherwise be an overwhelmingly complex world of possibilities. We must often rely on the heuristic, "What would I do in this situation?" In driving, for example, or playing a decent game of chess, tennis, or baseball, we rely constantly on a model of what others would do, which begins with a model of what we would do in their situation. There appears to be a physiological basis for this (Gallagher 2008). However, a highly competent tennis player will develop a model of an opponent, which is different from their self-model. The other player might be faster, or slower, or have a better backhand. Similarly, a competent development team will have mechanisms to realize that their end users are not "just like them." A full treatment of this topic is well beyond the scope of this chapter, but we speculate that providing tools to help facilitate both developing and using the human capacities for "Theory of Mind" tasks may offer a significant improvement in terms of lessening gratuitous complexity.

22.4 SOURCES OF COMPLEXITY IN THE DEVELOPMENT PROCESS

22.4.1 Radical Iteration in the Field

Although this topic is dealt with in more detail elsewhere in this handbook as well as in the article by Greene et al. (2003), it is worth at least a brief mention here that radical iteration in the field can generally help avoid, prevent, and address *all* the sources of complexity that we are about to enumerate. By working as closely as possible with the potential users of a system doing their real work in their real context, one can avoid unnecessary complexity that might otherwise be injected during problem finding, problem formulation, design, development, deployment, and maintenance.

22.4.2 Problem Finding

Most of our typical education focuses on solving problems that other people have already found and formulated. Great leverage can arise from finding and formulating problems. Conversely, unnecessary complexity in a system can be injected from the very beginning by "finding" a problem that is not really a problem (for the users). A computer scientist, for instance, might discover that users intuitively choose a sequence of tasks to complete that is not quite as efficient as the theoretically optimal sequence. The users choose routes that are "good enough" and "satisficing" (Simon 1962). However, the computer scientist (or accountant) might see the inefficiency as a "problem" that needs to be "solved." The result may be a system that requires users to key in a long list of tasks to be done, resources to be used, the data required by the tasks, and so on. The system that "solves" the (imaginary) problem could easily take far longer than the labor supposedly saved. Of course, there is the even more insidious problem that in order to model the efficiency of a process in the first place, certain simplifying assumptions must be made and these simplifications may well lead to a solution that is *less* nearly optimal in the real world than the original behavior. For example, in a large telecommunications company, a "route optimization" program was developed to dictate schedules to repair people. The program failed because it did not take into account many specific details that the repair people knew about the times and schedules and constraints of others with whom they had to coordinate.

Inadequate observation or starting with untested assumptions about what problems exist may also prevent people from even noticing easily solvable problems. On one factory tour in the early 1980s, one of the authors (JT) was shown someone whose job was to precisely align two silver needles. The person was sitting in a very awkward position and behind the silver needles was a silver background. It was explained to the author later that since the error rate was so high, they were working on a project to use machine vision to replace this position. The author pointed out that they might first try providing the person with ergonomic seating and a different background against which to align the needles. Automation proved unnecessary.

22.4.3 Problem Formulation and Requirements

One popular anecdote that illustrates the importance of proper problem formulation recounts a modern high-rise office building in which the office workers kept complaining about the slowness of the elevators. Computer programmers were called in, and they reprogrammed algorithms. The complaints increased. Engineers put in heavier duty cables and motors so that the elevators could move faster. Complaints remained as strong as ever and several important multifloor tenants threatened to move out. In desperation, the building owner was considering sacrificing some of the floor space and adding additional elevator shafts at what would obviously be a high price. Someone suggested putting mirrors on all the floors near the elevators and the complaints ceased. In this case, people initially assumed that the problem to be solved was that the elevators ran too slowly and focused on various ways to increase the speed. Eventually, someone came along who realized that the real problem was that people were *unhappy* about the elevator speed. Mirrors apparently gave them something to do and the time waiting did not seem so onerous.

In many real-world cases, this step is made more complex because typically there are a number of stakeholders, each of whom may have very different perceptions of the problem(s) to be solved. A sociotechnical pattern “Who Speaks for Wolf?” (Thomas, Danis, and Lee 2002) based on a Native American story transcribed by Paula Underwood (1993) suggests both the importance of finding all the relevant perspectives and stakeholders early in development and various techniques to try to accomplish that. Briefly, one of the members of a tribe was called “Wolf” because he made a life study of wolves. Once, while Wolf and a few other braves were on an extended hunting expedition, the tribe held council and decided they needed to move. A location was chosen and the tribe moved; however, a few months later, it became obvious that the tribe had moved into the midst of the spring breeding ground of the wolves. They had to decide whether to move again, post guards, or destroy the wolves. They finally decided to move again, but asked themselves, “What did we learn from this and how can we avoid this kind of error in the future?” Someone pointed out that if Wolf had been present at the first council, he would have advised against the location. From then on, they decided that whenever they made a major decision, they would ask themselves, “Who speaks for Wolf?” to see whether there were missing stakeholders or perspectives that needed to be taken into account. Many projects could profit from such a process.

It is typically well understood that the earlier in the process of development an error is caught, the less expensive it is. An error in problem formulation can be extremely long lived and expensive because it sets the context for measuring success. A product or system may be developed which appears to be successful at every step because the wrong thing is being measured so that ultimate failure goes unnoticed until too late. A “classic” example is the replacement of “Coca-Cola” with “New Coke.” According to Gladwell (2005), executives at the Coca-Cola Company were worried because they were losing some market share to Pepsi, and blind taste tests indicated that Pepsi was more often preferred. As a result, a sweeter version of Coke that tasted more like Pepsi was developed and put on the market as a replacement for Coke. The reaction was surprising, immediate, passionate, and nearly disastrous for Coke. People wanted the “old” Coke back! The entire story probably involves brand loyalty, memory, and cognitive dissonance, but one fundamental problem was that the Coca-Cola executives were assuming that the goal was to develop a new product that was sweeter so that it would be preferred in taste tests over Pepsi. And, they succeeded—at solving the wrong problem. What tastes best when you take a decontextualized sip is not necessarily what you prefer (for a variety of reasons, not all directly related to taste) day after day, month after month. In contrast to the “taste” test, the “case” test shows what people actually buy over a long period of time and of course, it is the latter measure which is actually important to profitability. In this case, the use of too *simple* a measure resulted in the wrong problem being solved. This is probably a common situation. Another example of this type occurred when the manufacturer for a new terminal for telephone operators did usability and productivity tests on operators using the terminal but failed to take into account the phone company *customer* who was also on the line and whose behavior actually turned out to be on the critical path most of the time. The entire design, development, testing, and so on were predicated on optimizing a system that consisted of the computer and the operator when the system that really mattered was the operator, the computer, and the *customer*. Again, an *initial* (over)-simplification resulted in a much more complex total solution than a more inclusive (and somewhat more complex) initial formulation would have produced. In this case, cognitive modeling (Gray, John, and Atwood 1993) subsequently confirmed by empirical tests helped avoid significant unnecessary expense and lead to redesign efforts.

Undue complexity can also be introduced by formulating the problem in overly complex terms. The idea of (and failure of) detailed centralized economic planning may be the quintessential example.

In summary, undue complexity can be introduced by beginning with the *wrong* problem formulation. Even if a better formulation is discovered later, unless the development team is willing to throw out everything and start over, remnants of the original formulation will tend to persist into design, development, deployment, testing, and so on, making both the resulting system and its associated elements (sales and marketing materials, documentation, education packages, problem determination aids, etc.) more complex than they need have been.

22.4.4 Design

In an attempt to break down a complex problem into manageable sub-problems, development teams, quite reasonably, divide into sub-teams to deal with various subsystems. Unfortunately, this can result in inconsistencies in basic functionality as well as in the user interface. For example, in one word processor, under certain conditions, the user would be faced with a message that said, in essence, "You cannot delete that file because it does not exist." However, an attempt to create another file by the same name resulted in a message that said, in essence, "You cannot create that file because it already exists." It seems clear that no commonly agreed upon definition of what it meant for a file to "exist" held sway through the whole of the development team.

At a more superficial level, one approach to reduce unnecessary discrepancies in the way that the user interface functions is to provide a "style guide" so that diverse developers or business partners working on various aspects or functions of an application suite will tend to provide a similar "look and feel." There is much to be said for this approach provided that it is applied with perspective and intelligence. In the worst case, development teams may blindly follow what was meant as a guideline and interpret it as an iron-clad rule. For instance, one of us (JT) became involved in the development of an application for the service representatives for a large telecommunications company. The corporate development team insisted that we "must" follow some "guidelines" that claimed users should choose an object before choosing an action. For users who must move between multiple applications, there is a relative advantage of having consistency among applications. In this specific context, however, the users did not use multiple applications, and it was clear that the way that they naturally thought of and interacted with the task, choosing an action first was far more intuitive, quicker, and less error prone than choosing an object first. Nonetheless, the management of the development team believed that the guidelines provided were "received truth" and therefore must be followed. (Incidentally, the project, which ultimately employed hundreds of programmers, never completed.)

22.4.5 Development

Perhaps the greatest contributor to undue complexity is that the development team (and to a lesser extent, management, associated marketing and sales, documentation specialists, etc.) becomes so familiar with a system that almost everything about it becomes "obvious" and "easy" regardless of how it might be perceived by an end user. Once one sees a hidden figure (such as a pig in the clouds), it is nearly impossible *not* to see it. Over the lifecycle of a product, tens, hundreds, or even thousands of little conventions and assumptions become second nature to those associated with the development.

The main cure for this malady is to continually test the scenarios of use, the designs, paper mock-ups, screenshots, prototypes, and beta versions with naive users, that is, folks who are representative of potential users but *not* part of the design team. If, for any reason, this is not feasible (e.g., security, the users do not yet exist) other techniques can provide some amelioration. For instance, heuristic evaluation (Nielsen and Landauer 1993) is likely to catch a fair proportion of actual problems. This is best done with HCI experts who are already familiar with both the technology and the application area. Typically, five to eight experts provide the most value. A variation on this technique involves having people successively take on different personae while interacting with a system, and this may increase the number of errors found in the same period (DeSurvire and Thomas 1993).

22.4.6 Testing

All too often, testing schedules become compressed. Partly as a result of time pressure and partly as a result of the fact that the development team has become accustomed to the high-level and low-level design, the error codes (or error messages) are often cryptic and designed for the productivity and convenience of the testing team or even left over from initial development. Unfortunately, in many cases, these error messages persist into the product experienced by the end user. Testers are often testing functions in a very well-specified context so that an error message is quite interpretable to them. However, to an end user, these same error messages are completely incomprehensible, and often the output comes from "dropping down" several layers in the software stack. Thus, an end user attempting to hit a button in a high-level application may see an error message which not only does not specify what corrective action to take; it mentions software elements that the end user did not even know existed.

Another issue with testing, of course, is that testers who work in the development organization may only test "reasonable" combinations of function. One of us (JT) designed the user experience for a "Dynamic Learning Environment" (Farrell et al. 2004; Thomas and Farrell 2006). In testing the code, he tried altering the URL returned by the system to find what he was looking for. The developers all knew that "you couldn't do this" and it caused the system to crash—but it is an often-used strategy. For instance, if you know that www.umich.edu/~person1 is the URL of person 1 and you are trying to find the website of their colleague at Michigan, person 2, you might reasonably suspect that their URL might be www.umich.edu/~person2 and indeed this often works. Happily, he was able to convince the developers to prevent this ploy from crashing the system. More generally, it is important to have potential *users* test the system as well as those steeped in the cultural assumptions of IT generally and a particular company or product specifically.

22.4.7 Deployment

Today, many applications include "Wizards" to help unzip, install, and even use a product. Often, the distribution of applications is via websites, and therefore finding the right application, finding the right version of the application, finding out whether one has all the prerequisites and necessary patches, and then deciding which features and options to turn on or off can prove as complex, or even more complex, than actually using the application. Often, the instructions and interfaces associated with deployment suffer obscurities from the same root cause as those introduced by the development team—being overly familiar with the application or system. For example, in recently attempting to download an upgrade to a system, the instructions cautioned the user to be sure to use one specific URL and not another. This URL, however, did not refer to the actual URL for finding the desired download. The required URL lay three layers down a menu structure in a *different application*. To the folks who deployed this update, the *context* in which this specific URL was to be used was so obvious as to not bother making that context explicit. A related problem occurs when paper documentation refers the user to look at a specific URL for important details. This generally works at the time the documentation is first released, but later, linking to that URL produces a "not found" error, and the user is referred to a very large generic website where the needed information, if it still exists at all, will be very troublesome to find.

22.4.8 Service

With the spread of an ever cheaper, higher bandwidth telecommunications infrastructure, service has experienced the dual trends of centralization of function and service being geographically distant from the users they are trying to support. Although centralization offers some benefits in economy of scale and knowledge sharing, having service personnel geographically distant can mean a decrease in shared context between service people and users they support. For example, local telephone operators used to be able to answer questions such as, "I need the number for that gas station across from the theater down town." Such questions are currently unanswerable from distant, centralized locations. In some cases, service personnel and users/customers may even have different native languages and come from different cultures. There is a possibility that some of this shared context may soon be reinstated, for example, from geographical information systems and from systems that allow service people to view screens and actions remotely. It may be that a shared view of the real visual world is unnecessary; a shared representation of the salient features, assuming they can be identified, may be enough (O'Neill et al. 2005).

From a user's perspective, undue complexity is often injected into the process of joint problem solving with service personnel because the service personnel are often organized according to the underlying system that they are trying to support rather than the symptom experienced by a user. An end user may, for instance, attempt to print something on a new printer and get an error message indicating that the print job failed. The user calls the "help desk" and the top level menu asks whether they are having a problem with connectivity, the operating system, or an application program, any of which may be the appropriate answer.

An apparently successful innovation in online help is the "Answer Garden" (Ackerman and Malone 1990), which allows expert users, over time with minimal disruption to "grow" a more and more detailed tree of FAQ's. Other trends making the end user's job of getting help simpler include the use of search engines as well as wiki's and blogs to support communities of practice and communities of interest.

22.4.9 Maintenance

Consider the maintenance of an automobile circa 1950. Although such machines were complex, the parts were all large enough to be visible and when a part broke, it could be replaced by a similar part. Such repairs might be simple or they might be complex in that replacing a part might require moving other parts to reach the part to be replaced. Modern software systems (and such systems are becoming ubiquitously embedded in all other technologies) introduce a set of new and often complex maintenance issues. Before installing a new upgraded version of one piece of software, for instance, it is often necessary to check for several prerequisite pieces of software. Each of these may in turn require still other prerequisites. Even if you already have one of the prerequisites installed; say, version XXX of program Y, you may also need to install a fair number of patches to XXX. In some cases, you need to uninstall software, taking care throughout to reboot as needed to clear out persistent memory. Indeed, even if this entire tree is followed, it may still be the case that installing some new piece of software causes something else that used to work, not to work any longer. Although the use of "wizards" has made maintenance much simpler in many cases, if something does fail, diagnosing and fixing the failure may require delving into several layers of software beneath the GUI that is supposed to provide the "simple" interface for the end user.

22.5 WAYS TO MEASURE COMPLEXITY

22.5.1 A Priori Mathematical Models

We have already mentioned an attempt to measure the complexity of programs by Halstead (1977). Other measures have been proposed; probably the best-known are those by McCabe (1976), McCabe and Butler (1989), and Jones (1996). These metrics aim to capture the inherent psychological complexity of the program and not just the behavior that the program surfaces to the end user. We might think to find useful analogues here to measures for the complexity of the user's interaction with a program as well. Unfortunately, the measures do not currently deal with several major factors of psychological complexity. First, the models do not deal with the complexity of the visual (or other sensory) stimuli presented to the user. Second, the models do not deal with the complexity of the response required of the user. Third, the models do not take into account the many complicating factors mentioned above that impact the correspondence between what is required of the user's behavior and what the user already knows.

One of the first attempts to quantitatively predict ease of use a priori tried to measure the complexity of the internal structure necessary to generate behavior (Reisner 1984). A similar approach has been used to try to predict learning time (Bovair, Kieras, and Polson 1990). Basically, both approaches consider the complexity of the underlying rule systems. Reisner's claim is that, other things being equal, it takes longer to execute a more complex rule set. Bovair, Kieras, and Polson claim that it takes longer to learn a more complex rule set and to execute it once learned. Both of these claims seem justified, though only with the caution of *ceteris paribus.* See Rautenberg (1996) for a meta-analysis of quantification attempts of user interfaces and Ivory and Hearst (2001) for a broader review of attempts to automate all or part of the process of usability evaluation.

22.5.2 Linear Regression

Psychological tests have often been constructed by taking a very large number of diverse items and then seeing which ones correlate with a desired criterion. Of course, one needs to revalidate these items with a new sample, but it is interesting that somewhat predictive tests have been constructed with this method for many domains. In fact, linear predictive models have been successfully applied to a large number of domains and are generally better than human experts (see, e.g., Dawes [1982]; Dawes, Faust, and Meehl [1989]). Walston and Felix (1977) used such a technique to predict the effort needed to complete a wide selection of software development projects. Interestingly, the most important factors had to do with the sociopolitical aspects of projects rather than the technical ones. A widely used, flexible method to predict various aspects of software development efforts is the Costructive Cost Model (COCOMO) (Boehm 1981, 2000). Although not strictly linear, the calculations are based on an empirical analysis of actual software development projects.

Perhaps such approaches can be profitably extended to psychological complexity as well. In fact, at least one attempt to measure the "goodness" (related at least in part to complexity) of websites used a similar approach (Ivory, Sinha, and Hearst 2001) and showed some evidence of success. However, in their study, somewhat different factors were predictive for different topic areas. Further, because conventions and underlying technology keep changing, it is likely that linear predictive equations might have to be updated on a frequent basis. In general, this is a potential limitation of linear predictive models. They can be quite accurate and quite robust but only provided the underlying conditions remain relatively constant. The absence of a theoretical underpinning can lead to predictions that can become inaccurate without warning.

22.5.3 Subjective Measures

An entirely different approach to measuring cognitive complexity is to use subjective measures; in effect, to ask people to rate or rank items in terms of cognitive complexity. In judgments of simple stimulus sequences, it appears that complexity and randomness are highly related (Falk and Konold 1997). An interesting and sophisticated attempt to provide a more differentiated subjective view of complexity applicable to real-world systems is that of "Cognitive Dimensions of Complexity" (Green 1989; Green and Petre 1996). This approach identifies 14 different dimensions of complexity. It can be used as a practical tool to focus the attention of developers successively on various areas of design that can lead to undue complexity. More recently, related work has focused on "mismatches" between the concepts inherent in the design of an artifact and the way users conceptualize (Connell, Blandford, and Green 2004). IBM's consumability measures aim at covering all aspects of the experience of different stakeholders and include both objective and subjective measures (Kessler and Sweitzer 2007; Calcaterra and Spangler, 2011).

22.5.4 Textual Analysis of Documentation

In general, we expect the complexity of a description of something to correlate highly with the complexity of that thing. Of course, it is possible to describe a simple thing with undue complexity, but in most circumstances, both extrinsic reward structures and the rules of conversation tend to push description to be sufficiently complex to describe something accurately but sufficiently simple to be understood. Therefore, one promising ersatz measure of the complexity of something, for example, a computer system or a procedure using that system, is to measure the complexity of the description. One commonly used measure is the "Flesch Formula" (Flesch 1948), which basically uses the average length of words in letters and the length of sentences in words in a document to provide as a score the educational reading level required to understand that document. Another metric is the "Gunning Fog" index (Gunning 1952), which seeks to measure the clarity of writing. A still more sophisticated automated approach is possible with the DICTION program (Hart 2001), which gives 31 primary and several derived dimensions to describe the rhetorical style of a document. One of the primary dimensions is labeled complexity and several of the others might also relate. Although this approach offers promise as a metric to be applied after the fact, caution must be applied to using it as an "in-process" metric. Otherwise, writers could use various tricks to make the description of an artifact appear "simple" while in reality making it incomprehensible. (In the absurd limiting case, one could provide a manual for a complex product that simply said, "Use it.") Clearly, documents must also be constrained to be *complete* in order for a fair and useful application of complexity metrics to descriptions.

22.5.5 Iterative Design and Testing

Clearly, much of the rest of this handbook expands on this topic. Although considerations of complexity may prove useful, at this point, there is no complete substitute for interacting with real users doing real work in their real context as a general method for designing, developing, and deploying truly useful and usable systems. Thinking about and measuring complexity may help in this process, however, especially in cases where real user testing is impractical.

22.6 POSSIBLE FUTURE APPROACHES

One can imagine that ever more sophisticated techniques for brain activity imaging and other physiological measures may someday render a reliable and easy-to-use objective index of psychological complexity. Somewhat more likely is the continued evolution of modeling approaches such as Soar (Laird, Newell, and Rosenbloom 1987) Executive Process Ineractive Control [EPIC] (Kieras and Meyer 1997) CogTool (John et al. 2004), Adaptive Control of Though-Rational [ACT-R] (Taatgen and Anderson 2008), and Information Foraging (Pirolli 2007). The follow-on programs might someday conceivably "automatically" calculate a set of complexity numbers for combinations of user, task, system, and context. In the nearer term, it seems likely that continued improvement of heuristic approaches that do require some human intervention will prove useful (e.g., "Dimensions of Cognitive Complexity" [Green 1989] and the Brown, Keller, and Hellerstein [2005] model) in helping to design better user experiences. The latter model assumes that an "expert path" is known and complexity depends on the number of actions, the memory load associated with retaining needed parameters, and the number of context shifts.

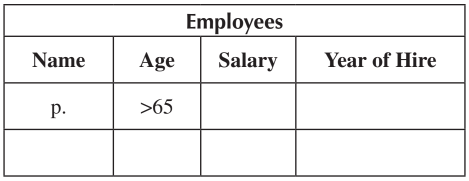
In our own work, we are attempting to extend this model to account more completely for points of uncertainty and the difficulties of decision making. In addition, we are building tools that enable complexity metrics to be produced as a side effect of normal development processes. In this way, developers may gain feedback that is both timelier and more differentiated with respect to their design decisions. In the short term, such metrics should provide useful feedback. As the tools come to be used over time, we expect the development teams to internalize the implications so that better initial design decisions are made.

22.7 COMPLEXITY REDUCTION IN PRACTICE: CASE STUDIES

Clearly, complexity is itself a complex topic. But it is clearly not intractable. Numerous examples of good design and development leading to suitable levels of complexity may be found. We conclude this chapter with five case studies from our own laboratory that illustrate how complexity may be considerably reduced in a real-world setting.

22.7.1 Case Study 1: Query By Example

Query By Example, invented by Moshe Zloof in the early 1970s, was an attempt to provide a query language easy enough for computer-naive novices to learn to use. Early studies indicated that nonprogramming high school and college students were fairly successful in translating English questions into the syntax of this formal query language (Thomas and Gould 1975). One probable reason for the relatively fast times and low error rates was that, in many cases, users only had to *select* the right place to write information rather than having to write all information "from scratch." Another likely reason was that, in Halstead's (1977) terms, Query By Example was quite "dense" (see Figure 22.10). That is, almost all of the symbols required for a given query were symbols that would be required in any conceivable query language. There was little overhead. In other words, the intrinsic complexity of any given query remained, but there was no "gratuitous" or undue complexity introduced by the system itself.



Employees			
Name	**Age**	**Salary**	**Year of Hire**
p.	>65		

"Print the names of everyone who is more then 65 years of age." Only the "p." and the ">65" is entered by the user.

FIGURE 22.10 A simple query in Query By Example.

In some cases, students continued to have some difficulties, but further studies showed that these difficulties did not spring from the query language per se, but from general difficulties with logic (Thomas 1976). For example, students had some difficulties with quantifiers (all vs. some), logical connectives (and, or, not) and with operational distinctions (sum, count, and count unique). These same difficulties occurred in the absence of any requirements of the query language. Additional difficulties were encountered when the structure of the data base did not match well with the structure of the question. So, for instance, if the database contained a column labeled "Year of Hire" a question asking for the names of people "hired after 1970" was fairly easy. A question asking for the names of people "who have worked for more than 35 years" however, was much more problematic.

In a follow-on study (Thomas 1983), students were not given English questions to translate but instead were shown a database structure and given problem statements. They were then to generate their *own* English questions relevant to the problem and then translate their own questions into Query By Example. These results were much less encouraging. Basically, these students had very little concept of what types of questions were reasonable to ask the computer. One problem given was, "Some of the younger faculty members feel that they are not paid enough relative to the older faculty. Write a question that you think might shed light on this issue." A typical question was, "Are the younger faculty paid enough?" In other words, many of the students expected the computer to simply "formulate and solve the problem" for them.

In the 1970s, novice computer users not only might have trouble writing a correct query; they might also write an incorrect query that would consume a huge amount of resource. For this reason, it was considered worthwhile to try to predict the chances that a query was correct before submitting it to the search engine. Based on the formal properties of the query alone (number of operators, etc.), we were able to predict 54% of the variance in query accuracy. If we added the time taken to write the query and the student's own confidence rating, we achieved an accuracy of 75%. Finally, if we added a term for whether the mapping between the English and query was straightforward, more than 90% of the variance was predictable. All these measures may be thought of as factors related to psychological complexity. Although the interface and language were developed and improved iteratively, our studies also showed that remaining user difficulties were due to intrinsic task complexity.

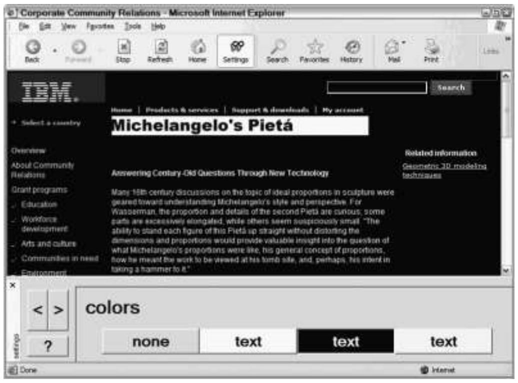
22.7.2 Case Study 2: Web Accessibility Technology

With the web growing in importance to both recreation and work, it became clear to us some time ago that barriers to web use needed to be lowered. It was no longer acceptable for people with visual and motor disabilities to be denied effective web access.

We began our explorations in this area by working directly with older adults, a population needing both web content modifications (text and image enlargement, navigational simplification, visual contrast modifications, etc.) and input adaptations (to filter out hand tremors, remove extraneous key presses and mouse clicks, etc.). Of course, modern browsers and operating systems provide a number of these modifications and adaptations but access to them is distributed throughout the system, inconsistently surfaced, and, often inaccessible to those most in need. Our challenge lay in finding a way to simultaneously reduce the complexity of so much functionality while adding still more accessibility features not otherwise available.

Our approach was iterative and user centered. Early on it became clear that our users did not want a "simplified" browser. Nor did they want to be restricted to a subset of the web. After multiple attempts (chronicled in the work by Richards and Hanson [2004]; Hanson and Richards [2005]), we converged on a design that added a set of very simple control panels to the bottom of an otherwise normal browser, each control panel being dedicated to one class of adaptation or transformation. See Figure 22.11 for one such panel in the context of the full browser.

The only change to the browser's normal interface is the addition of a single "Settings" button. Clicking on this button brings up the first of the series of control panels. Clicking on one of the buttons in a control panel applies an immediate change to the web content. If the change is desired, it is kept by doing nothing more with the panel. Otherwise, it is cancelled by clicking on a "None" or "Standard" button depending on the panel. The panels are organized in a conceptual ring. Clicking on the > button brings up the next panel. Clicking on the < button brings up the previous panel. Clicking on the help button brings up an interactive help page keyed to the current panel, which allows the user to easily explore the full range of modification the panel exposes.



Michelangelo's Pietá

Answering Century-Old Questions Through New Technology

Many 16th century discussions on the topic of ideal proportions in sculpture were geared toward understanding Michelangelo's style and perspective. For Wasserman, the proportion and details of the second Pietá are curious; some parts are excessively elongated, while others seem suspiciously small. "The ability to stand each figure of this Pietá up straight without distorting the dimensions and proportions would provide valuable insight into the question of what Michelangelo's proportions were like, his general concept of proportions, how he meant the work to be viewed at his tomb site, and, perhaps, his intent in taking a hammer to it."

FIGURE 22.11 The "colors" panel for web accessibility software.

A number of mechanisms lie beneath this simple surface. Some modifications are applied by altering browser settings stored in the systems global registry. Others are applied by automatically creating and installing a user style sheet. Others are applied by adjusting parameters in the underlying accessibility settings of the system. Others are applied by modifying the web page's in-memory Document Object Model. Still others are applied by means of always running agents examining input event streams. All of this complexity is hidden from the user.

For a concrete example of the resulting simplification, consider the case of changing the colors of web pages. This is a modification often found useful for people with visual disabilities and for people with various forms of dyslexia. To change just the text foreground and background colors on a popular browser requires 16 separate steps. To ensure that link, visited link, and hover colors are suitably contrasting requires 24 additional steps. In contrast, in our design, setting two of the most popular color combinations—black text on white backgrounds or white text on black backgrounds—requires only a single click on a button showing the desired color combination. Optimally contrasting link, visited link, and hover colors are automatically set as a side effect of this choice. For those needing a broader range of choice, another simple panel allows the direct setting of the foreground and background RGB values. Again, the link, visited link, and hover colors are set automatically.

This case study illustrates a number of approaches to reducing software complexity. First, iterative user-centered design allowed for the rapid exploration of a range of alternative interfaces. Second, the addition of a new interface layer allowed for the unification of a number of disparate mechanisms. Third, careful analysis of the needs of users allowed for some choices to be surfaced prominently and others to be removed. Fourth, the provision of immediate feedback allowed users to directly experience the effects of any choice, trying them alone or in combination until their overall web experience was optimal.

22.7.3 Case Study 3: Teaching Better Strategies

Users are often too *busy* with the task at hand to try to *optimize* their behavior. When they are using tools, the attention of the expert user is likely to be on the goal, and the progress toward the goal and not on the *processes* that they are using to achieve that goal. In addition, it may not be at all clear that their current behavior is significantly less than optimal. This is likely to be particularly true for knowledge workers today because many people work on individual and unique tasks in isolation. In the past, hunters, gatherers, farmers, and widget makers would often have quite similar tasks and quite objective means for determining their performance relative to their peers. In today's world, this remains true in some domains, for example, sports. It is quite common for people to compare their performance on comparable tasks such as the time to run a specific distance or the score on a particular golf course. However, suppose that an athlete never directly observed another one doing the same task nor had information about performance on identical tasks. A person might well run an 8-minute mile or shoot 110 for 18 holes and believe these to be quite adequate performances.

Even if a person does realize that their performance can and should be improved, it is still a complex *and different task* to determine *how* their strategy must change to enhance their performance. Indeed, it may not even be obvious that *strategy* as opposed to *execution* is a major block to better performance. The complexity of determining attribution is illustrated by the work already mentioned on what might *seem* to be a simple athletic skill, namely, putting. An early example of the impact of strategy on performance comes from studies done in our laboratory on problem solving (Carroll, Thomas, and Malhotra 1980). Left to be independent, subjects quite naturally represented a spatial problem in terms of a spatial layout and performed relatively well. Given an isomorphic problem stated in terms of scheduling, however, subjects did relatively poorly and did not use a spatial representation. When encouraged to do so, they were able to use such a representation and the performance differences disappeared. In other words, while the objective complexity of the task remained constant, the psychological complexity was mediated by the representational strategy chosen and people did not automatically choose a good strategy. Earlier work (Thomas 1974) showed that virtually noone approached a simple river-crossing problem in terms of a strategy guaranteed to work; viz., exhaustively exploring the rather small problem graph. Instead, subjects appeared to want to insist on continually making "progress" toward their goal when the solution requires what appears superficially to be moving away from the goal.

In the cases reported above, subjects were presented with novel experimental tasks. However, Bhavnani and John (2001) examined the behavior of domain experts (in architectural design) who had experience with a computer-aided design system and still found that they used nonoptimal strategies. Bhavnani, Reif, and John (2001) extended this work and found that students taught effective strategies and functionality (in an equal time period) performed better than students whose instruction focused only on functionality. In further related work, Bhavnani et al. (2006) found that it is possible to design tools to help guide users into using a more fruitful search strategy when searching for medical information.

Another example of strategy support comes from the domain of storytelling (Landry and Guzdial 2006). Although storytelling is a "natural" and effective way to handle complex events, communicate and build social capital, (Thomas 1999) almost no public school training is typically given on the effective strategies for narrative construction, at least in the United States. Landry and Guzdial recognized people's desire to tell stories and built a plug-in for Flickr to allow them to construct stories around personal photographs. However, some strategy support was necessary to help make these stories more effective.

The lessons to be learned from this and related research are threefold: (1) people do not necessarily hit on strategies to simplify their tasks, (2) better strategies can be taught, and (3) tools may help guide people into using better strategies.

22.7.4 Case Study 4: Coping with Ecological Complexity

An important new area of research in HCI is the application of HCI principles to helping deal with ecological issues. One aspect of this, that our laboratory has been particularly interested in, is the use of *human intelligence* in addressing ecological challenges. The ecological domain is intrinsically a very complex one and many attempts to improve sustainability are being made in terms of top-down optimizations. However, we believe that in many cases, this can be complemented by making use of citizen knowledge. Citizens are often very motivated to deal with local issues, and have detailed knowledge specific to time and place. With the ubiquity of computing technology and sensors, there are increased possibilities for inputting huge quantities of data relevant to a particular problem. An interesting example of the envisioned genre is the Cyclopath application (Panciera et al. 2010), which allows bicyclists in the Minneapolis area to view and input information about the bikeability of various routes.

A related thread of work examines the role of distributed intelligence in the handling of complexity of disasters (e.g., Vieweg et al. 2010; Palen and Liu 2007). When disasters strike, officials attempt to gather, organize, and disseminate data to reduce the human impact of these disasters. Unfortunately, for a number of reasons, studies of such incidents as the shootings at Virginia Tech, the Katrina Hurricane, and wildfires in California indicate that "official" information is often inaccurate or not timely. With the widespread use of mobile phones, people attempt to gather, organize, and disseminate timely and useful information themselves. This has the potential, on the one hand, of providing more locally useful and timely information but on the other hand of spreading unfounded rumors.

Work in our laboratory is underway on a similar topic, namely, to allow more open communication between a city command center and the citizenry. City command centers offer another example of complex work requiring the coordination of many sources of information as well as the expertise of many people. The work on the human factors of the command centers is excellent but historically does tend to treat such centers as self-contained entities. In the past, this was generally a valid simplifying assumption but in today's world, there are both significant opportunities and challenges from understanding the role that crowd-sourcing may play in such command centers. The lesson to be learned from this line of research is that one way of helping to deal with complex real-world situations is to design for input from a large number of people and provide natural ways to vet and organize that information.

22.7.5 Case Study 5: Productivity Tools for High-Performance Computing

As part of a long-term project, IBM is building a new generation of super-computers (http://www.research.ibm.com/hptools/). We would all like to see these computers evidence not only greater throughput but greater programmer productivity as well. Programming is typically a quintessentially complex task and much research has been conducted to attempt to model programmer productivity, to make "programming" accessible to nonprogrammers, and to provide tools to enhance programmer productivity. It seems clear that factors other than tools and task complexity can have a major impact on individual and team productivity (Brooks 1975; Walston and Felix 1977; DeMarco and Lister 1987). Nonetheless, it seems clear that various tools can also contribute substantially to reducing the *undue* complexity in this domain. This is particularly important in high-performance computing. Although programming itself is always complex, *parallel* programming typical of high-performance computing can be *particularly* complex.

IBM is attempting to enhance programmer productivity through the development of a suite of tools, through the encouragement of further open source tools built on Eclipse, and with a new programming language, X10 (http:x10-lang.org). Part of our group is measuring the productivity gains that such tools provide compared with people attempting to do similar tasks in 2002. Purely objective empirical comparisons are problematic, both because there are no detailed baseline data from 2002 and because there are as yet no experts in the tools, which are still under development.

One of several approaches we are using, therefore, is to *model* the complexity of the tasks required by the 2002 tools that were available and to compare that with models of the tasks required by the new tools. We assume that the parameters of the human information processor are unchanged and that in both cases, expert behavior eventually emerges, is taught, or developed through sharing within a community of practice. This is one example of the "time machine" aspect of cognitive modeling since it allows comparisons between something that did but no longer exists and something that does not yet exist! Our earlier modeling efforts were based on the model of Brown and Hellerstein (2004), whereas more recent efforts have been rendered in the keystroke level models first explicated in Card, Moran, and Newell (1983) but now much more conveniently done with the advent of CogTool (Bellamy et al. 2010; Richards et al. 2010). Further discussion of the history and usage of cognitive modeling tools can be found in Kieras (2012). In combination with empirical evaluations, retrospective analyses, and other methods, we believe we can reach a reasonable understanding of the savings in undue complexity due to the new suite of tools.

22.8 CONCLUSIONS

Not surprisingly, psychological complexity is itself a complex topic. It should be of interest to designers of HCI because they typically need to reduce undue complexity. There are occasional exceptions, for instance, in entertainment and learning applications where the designer may want to *increase* psychological complexity. In this chapter, we have distinguished complexity from many related but different concepts such as uncertainty. We have also explored how undue complexity may be introduced into the development process and suggested approaches to measure and reduce undue complexity. As the intrinsic complexity of the world seems to be increasing, the importance of reducing undue complexity will continue to increase.

REFERENCES

Abley, M. 2005. *Spoken Here: Travels Among Threatened Languages*. London: Arrow Books.

Ackerman, M., and T. Malone. 1990. Answer Garden: A Tool for Growing Organizational Memory. *In Proceedings of the ACM Conference on Office Information Systems*, 31–9, April 1990. Cambridge, MA.

Ahlberg, C., C. Williamson, and B. Shneiderman. 1992. Dynamic queries for information exploration: An implementation and evaluation. In *Proceedings of the ACM CHI'92: Human Factors in Computing Systems*, 619–26. New York: ACM.

Alexander, C. A. 2002. *The Nature of Order: An Essay on the Art of Building and the Nature of the Universe. Book One: The phenomenon of Life*. Berkely, CA: Center for Environmental Structure.

Attneave, F. 1957. Physical determinants of the judged complexity of shape. *J Exp Psychol* 53:221–27.

Barlett, F. C. 1932. *Remembering: An Experimental and Social Study*. Cambridge: Cambridge University Press.

Bar-Yam, Y. 1997. *Dynamics of complex systems (Studies in nonlinearity)*. Boulder, CO: Westview Press.

Bar-Yam, Y. 2000. *Unifying Themes in Complex Systems: Proceedings of the International Conference on Complex Systems*. New York: Basic Books.

Bass, L., and B. E. John. 2003. Linking usability to software architecture patterns through general scenarios. *J Syst Softw* 66:187–97.

Bellamy, R., B. John, J. Richards, and J. Thomas. 2010. Using CogTool to model programming tasks. In *PLATEAU Workshop at ONWARD/SPLASH 2010*, Reno, Nevada, November. Article 1, doi>10.1145/1937117.1937118.

Bhavnani, S. K., C. K. Bichakjian, T. M. Johnson, R. J. Little, F. A. Peck, J. L. Schwartz, and V. J. Strecher. 2006. Strategy hubs: Domain portals to help find comprehensive information. *J Am Soc Inf Sci Technol* 57(1):4–24.

Bhavnani, S. K., and B. E. John. 2001. The strategic use of complex computer systems. In *Human-Computer Interaction in the New Millennium*, ed. J. M. Carroll. Reading, MA: Addison-Wesley/ACM Books.

Bhavnani, S. K., F. Reif, and B. E. John. 2001. Beyond command knowledge: Identifying and teaching strategic knowledge for using complex computer applications. In *Proceedings of CHI 2001*, 229–36. New York: ACM.

Boehm, B. W. 1981. *Software Engineering Economics*. Upper Saddle River, NJ: Prentice Hall.

Boehm, B. W. 2000. *Software Cost Estimation with COCOMO 2000*. Upper Saddle River, NJ: Prentice Hall.

Bovair, S., D. E. Kieras, and P. G. Polson. 1990. The acquisition and performance of text-editing skill: A cognitive complexity analysis. *Hum Comput Interact* 5(1):1–48.

Bransford, J. D., and M. K. Johnson. 1973. Considerations of some problems in comprehension. In *Visual Information Processing*, ed. W. G. Chase. New York: Academic Press.

Brooks, F. P. 1975. *The Mythical Man-Month: Essays on Software Engineering*. Reading, Boston, MA: Addison-Wesley.

Brown, G. D. A. 1984. A frequency count of 190,000 words in the London-Lund Corpus of English Conversation. *Behav Res Methods Instrum Comput* 16:502–32.

Brown, A. B., and J. L. Hellerstein. 2004. An approach to benchmarking configuration complexity. In *Proceedings of the 11th ACM SIGOPS European Workshop*, September, 2004. NY: ACM. Article 18, doi>10.1145/1133572.1133609.

Brown, A. B., A. Keller, and J. L. Hellerstein. 2005. A model of configuration complexity and its application to a change management system. In *Proceedings of the Ninth IFIP/IEE International Symposium on Integrated Network Management, IM 2005*, 634–44, May 2005. New York: IEEE.

Bruckman, A. S. 1997. Moose crossing: construction, community, and learning in a networked virtual world for kids. Doctoral Thesis. UMI Order Number: AAI0598541.

Calcaterra, J. A., and D. Spangler. 2011. Designing for learnability in user interface migrations: Experiences with designing a serviceability user interface. In *Proceedings of the 5th Annual Symposium on Computer Human Interaction for Management of Information Technology*, 469–88. New York: ACM.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Erlbaum.

Carroll, J., and C. Carrithers. 1984. Training wheels in a user interface. *Commun ACM* 27(8):800–06.

Carroll, J., and R. L. Mack. 1984. Learning to use a word processor by doing, by thinking, and by knowing. In *Human Factors in Computer Systems*, ed. J. C. Thomas and M. L. Schneider. Norwood, NJ: Ablex.

Carroll, J. B., and M. N. White. 1993. Word frequency and age of acquisition as determiners of picture-naming latency. *Quarterly journal of experimental psychology* 25:85–95.

Carroll, J., J. C. Thomas, and A. Malhotra. 1980. Presentation and representation in design problem solving. *Br J Psychol* 71(1):143–55.

Chernoff, H. 1973. The use of faces to represent points in k-dimensional space graphically. *J Am Stat Soc* 68:361–68.

Connell, I., A. Blandford, and T. R. G. Green. 2004. CASSM and cognitive walkthrough: Usability issues with ticket vending machines. *Behav Inf Technol* 23(5):307–20.

Coskun, E., and M. Grabowski. 2005. Impacts of user interface complexity on user acceptance and performance in safety-critical systems. *Emerg Manag* 2(1):1–29.

Dawes, R. 1982. The robust beauty of improper linear models in decision-making. In *Judgment under Uncertainty: Heuristics and Biases*, ed. D. Kahneman, P. Slovic, and A. Tversky, 391–407. Cambridge: Cambridge University Press.

Dawes, R., D. Faust, and P. Meehl. 1989. Clinical versus actuarial judgment. *Science* 241:1668–74.

DeGroot, A. D. 1978. *Thought and Choice in Chess*. 2nd ed. The Hague, The Netherlands: Mouton.

DeMarco, T., and T. Lister. 1987. *Peopleware: Productive Projects and Teams*. New York: Dorset.

den Ouden, E. 2006. Development of a Design Analysis Model for Consumer Complaints. Doctoral Dissertation. Technical University of Eindhoven.

Desurvire, H., and J. C. Thomas. 1993. Enhancing the performance of interface evaluation using non-empirical usability methods. In *Proceedings of the 37th Annual Meeting of the Human Factors and Ergonomics Society*, 1132–6. Santa Monica, CA: Human Factors & Ergonomic Society.

Ellis, A. 2001. *Overcoming Destructive Beliefs, Feelings and Behaviors: New Directions in Rational Emotive Behavior Therapy*. New York: Albert Ellis Institute.

Farrell, R., J. Thomas, B. Rubin, D. Gordin, A. Katriel, R. O'Donnell, E. Fuller, and S. Rolando. 2004. Personalized just-in-time dynamic assembly of learning objects. In eds. J. Nall & R. Robson, *Proceedings of the world conference on E-learning in corporate, government, healthcare, and higher education*, 607–14. Chesapeake VA: AACE.

Flesch R. F. 1948. A new readabilitiy yardstick. *J Appl Psychol* 32:221–33.

Fozard, J. L., J. C. Thomas, and N. C. Waugh. 1976. Effects of age and frequency of stimulus repetitions on two-choice reaction time. *J Gerontol* 31(5):556–63.

Furnas, G. W., T. K. Landauer, L. M. Gomez, and S. Dumais. 1984. Statistical semantics: Analysis of the potential performance of keyword information systems. In *Human Factors in Computing Systems*, ed. J. C. Thomas and M. L. Schneider, 187–242. Norwood, NJ: Ablex.

Gallagher, S. 2008. Neural simulation and social cognition. In *Mirror Neuron Systems: The Role of Mirroring Processes in Social Cognition*, ed. J. A. Pineda, 355–71. Totowa, NJ: Humana Press.

Gladwell, M. 2005. *Blink: The Power of Thinking without Thinking*. New York: Little Brown.

Gonzalez, V. M., and G. Mark. 2005. Managing currents of work: multi-tasking among multiple collaborations. In *Proceedings of the Ninth European Conference on Computer-Supported Cooperative Work, ECSCW 2005*, 113–20. The Netherlands: Springer.

Gould, J. D., and S. J. Boies. 1984. Speech filing: An office system for principals. *IBM Syst J* 23(1):65–81.

Gray, W. D., B. E. John, and M. E. Atwood. 1993. Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Hum Comput Interact* 8:237–309.

Gray, W. D., B. E. John, R. Stuart, D. Lawrence, and M. E. Atwood. 1990. GOMS meets the phone company: Analytic modeling applied to real-world problems. In *Proceedings of IFIP Interact'90: Hum-Comput Interact*, 29–34. Amsterdam: North Holland.

Green, T. G. R. 1989. Cognitive dimensions of notations. In *People and Computers V*, ed. A. Sutcliffe and L. Macaulay. Cambridge: Cambridge University Press.

Green, T. G. R., and M. Petre. 1996. Usability analysis of visual programming environments: A cognitive dimensions approach. *J Vis Lang Comput*. 7:131–74.

Greene, S. L., L. Jones, P. Matchen, and J. C. Thomas. 2003. Iterative development in the field. *IBM Syst J* 42(4):594–612.

Grinter, R. E., and W. K. Edwards. 2005. The work to make a home network work. In *Proceedings of the Ninth European Conference on Computer-Supported Cooperative Work, ECSCW 2005*, 469–88. The Netherlands: Springer.

Gunning, R. 1952. *The Technique of Clear Writing*. New York: McGraw-Hill International Book Co.

Halstead, M. H. 1977. *Elements of Software Science*. New York: Elsevier North-Holland.

Hanson, V. L., and J. T. Richards. 2005. Achieving a more usable World Wide Web. *Behav Inf Technol* 24(3):231–46.

Harris, B. N., B. E. John, and J. Brezin. 2010. Human performance modeling for all: Importing UI prototypes into CogTool. In *Proceedings of the 28th of the international Conference Extended Abstracts on Human Factors in Computing Systems (Atlanta, Georgia, USA, April 10–15, 2010), CHI EA '10*, 3481–86. New York: ACM.

Hart, R. P. 2001. Redeveloping diction: Theoretical considerations. In *Theory, Method and Practice in Computer Content Analysis*, ed. M. West. Westport, CT: Ablex.

Hildebrandt, N., D. Caplan, S. Sokol, and L. Torreano. 1995. Lexical factors in the word-superiority effect. *Mem Cogn* 23(1):23–33.

Holland, J. J. 1995. *Hidden Order: How Adaptation Builds Complexity*. New York: Basic Books.

Ivory, M. Y., and M. A. Hearst. 2001. The state of the art in automating usability evaluation of user interfaces. *ACM Comput Surv* 33(4):470–516.

Ivory, M. Y., R. R. Sinha, and M. A. Hearst. 2001. Empirically validated web page design metrics. In *Proceedings of CHI' 01*, 53–60. New York: ACM.

John, B. E. 1990. Extensions of GOMS analyses to expert performance requiring perception of dynamic visual and auditory information. In *Proceedings of CHI'90*, 107–15. New York: ACM.

John, B. E., L. Bass, E. Golden, and P. Stoll. 2009. A responsibility-based pattern language for usability-supporting architectural patterns. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing*, Pittsburgh, PA, July 15–17, 2009. New York: ACM.

John, B. E., K. Prevas, D. D. Salvucci, and K. Koedinger. 2004. Predictive human performance modeling made easy. In *Proceedings of CHI 2004*, Vienna, Austria, April 2004, 455–62. New York: ACM.

Johnson, M. H., and J. Morton. 1991. *Biology and Cognitive Development: The Case of Face Recognition*. Oxford, UK and Cambridge, MA: Blackwell.

Jones, C. 1996. *Applied Software Measurement*. New York: McGraw-Hill.

Kessler, C., and J. Sweitzer. 2007. *Outside-In Software Development: A Practical Approach to Building Successful Stakeholder-based Products*. Indianapolis, IN: IBM Press.

Kidd, C., S. T. Piantadosi, and R. N. Aslin. 2010. The Goldilocks effect: Infants' preference for stimuli that are neither too predictable nor too surprising. In *Proceedings of the 32nd Annual Meeting of the Cognitive Science Society*.

Kieras, D. 2012. Model-based evaluation. In *The Human-Computer Interaction Handbook*. 3rd ed., ed. J. A. Jacko, Chapter 57. New York: Erlbaum.

Kieras, D., and D. E. Meyer. 1997. An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Hum Comput Interact* 12:391–438.

Klinger, A., and N. A. Salingaros. 2000. A pattern measure. *Environ Plann B Plann Des* 27:537–47.

Kucera, H., and W. N. Francis. 1967. *Computational Analysis of Present-day American English*. Providence, RI: Brown University Press.

Laird, J. E., A. Newell, and P. S. Rosenbloom. 1987. SOAR: An architecture for general intelligence. *Artif Intell* 33:1–64.

Landry, B. M., and M. Guzdial. Learning from Human Support: Informing the Design of Personal Digital Story-Authoring Tools. In *iDMAa + IMS Conference: CODE*, (2006), Available at http://www.units.muohio.edu/codeconference/proceedings/conference_papers4.htm.

Lewis, C. 2007. Simplicity in cognitive assistive technology: A framework and agenda for research. *Univers Access Inf Soc* 5(4):351–61.

McCabe, T. J. 1976. A Complexity Measure. *IEEE Trans Softw Eng* SE-2(4):308–20.

McCabe, T. J., and C. W. Butler. 1989. Design complexity measurement and testing. *Commun ACM* 32(12):1415–25.

Miller, G. A. 1956. The magic number seven, plus or minus two: Some limits on our capacity for processing information. *Psychol Rev* 63: 81–97.

Myers, B. A., R. G. McDaniel, and D. S. Kosbie. 1993. Marquise: Creating complete user interfaces by demonstration. In *Proceedings of INTERCHI'93: Human Factors in Computing Systems,* 293–300, April 24–29. New York: ACM.

Nielsen, J., and T. K. Landauer. 1993. A mathematical model of the finding of usability problems. In *Proceedings of ACM INTERCHI '93*, 206–213. New York: ACM Press.

Norman, D. A. 1988. *The Psychology of Everyday Things*. New York: Basic Books.

O'Neill, J., S. Castellani, A. Grasso, F. Roulland, and P. Tolmie. 2005. Representations can be good enough. In *Proceedings of Ninth European Conference on Computer-Supported Cooperative Work, ECSCW 2005*, 267–86. The Netherlands: Springer.

Paivio, A., J. C. Yuille, and S. A. Madigan. 1968. Concreteness, imagery and meaningfulness values for 925 words. *J Exp Psychol Monogr Suppl* 76(3, part 2).

Panciera, K., R. Priedhorsky, T. Erickson, and L. Terveen. 2010. Lurking? Cyclopaths? A quantitative lifecycle: Analysis of user behavior in a geowiki. In *Proceedings of CHI 2010*, 1917–26. New York: ACM Press.

Palen, L., and S. Liu. 2007 Citizen communications in crisis: Anticipating a future of ICT. In *Proceedings of ACM Conference on Human Factors in Computing Systems.* 727–36. NY: ACM.

Papert, S. 1993. *The children's Machine: Rethinking School in the Age of the Computer*. New York: Basic Books.

Pelz, D. 2000. *Dave Pelz's Putting Bible*. New York: Doubleday.

Pickover, C. 1985. Tusk: A versatile graphics workstation for speech research. IBM Research Report: RC 11497.

Pirolli, P. 2007. *Information Foraging Theory: Adaptive Interaction with Information*. New York: Oxford University Press.

Premack, D. G., and G. Woodruff. 1978. Does the chimpanzee have a theory of mind? *Behav Brain Sci* 1:515–26.

Pressing, J. 1999. Cognitive complexity and the structure of musical patterns. In *Proceedings of the 4th Conference of the Australasian Cognitive Science Society*.

Reicher, G. M. 1969. Perceptual recognition as a function of meaningfulness of stimulus material. *J Exp Psychol* 81:275–80.

Reisner, P. 1984. Formal grammar as a tool for analyzing ease of use: Some fundamental concepts. In *Human Factors in Computer Systems*, ed. J. C. Thomas and M. L. Schneider, 53–78. Norwood, NJ: Ablex.

Resnick, M., J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner et al. 2009. Scratch: Programming for all. *Commun ACM* 52(11):60–7.

Richards, J., R. Bellamy, B. John, C. Swart, and J. C. Thomas. 2010. Using CogTool to Model Programming Tasks, Psychology of Programming Interest Group: Work in Progress.

Richards, J. T., and V. L. Hanson. 2004. Web accessibility: A broader view. *In Proceedings of the Thirteenth International ACM World Wide Web Conference, WWW2004*. New York: ACM.

Roget's New Millennium™ *Thesaurus.* 2005. 1st ed. (v 1.1.1) San Antonio, TX: Lexico Publishing Group, LLC.

Rubin, D. C. 1980. 51 Properties of 125 words: A unit analysis of verbal behavior. *J Verb Learn Verb Behav* 19:736–55.

Shannon, C. E., and W. Weaver. 1949. *The Mathematical Theory of Communication*. Urbana, IL: University of Illinois Press.

Shmulevich, I., and D. J. L. Povel. 2000. Complexity measures of musical rhythms. In *Rhythm Perception and Production*, ed. P. W. M. Desain and W. L. Windsor, 239–44. Lisse, the Netherlands: Swets & Zeitlinger (Studies on new music research, 3).

Shmulevich, I., O. Yli-Harja, E. Coyle, D. Povel, and K. Lemstrom. 2001. Perceptual issues in music pattern recognition: complexity of rhythm and key finding. *Comput hum* 35:23–35.

Simon, H. A. 1962. The architecture of complexity. *Proc Am Philos Soc* 106:467–82.

Simon, H. A., and M. Barenfeld. 1969. Information processing analysis of perceptual processes in problem solving. *Psychol Rev* 76:473–83.

Simon, H. A., and K. J. Gilmartin. 1973. A simulation of memory for chess positions. *Cogn Psychol* 5:29–46.

Smith, M. A., G. W. Cottrell, and K. L. Anderson. 2001. The early word catches the weights. In *Advances in Neural Information Processing Systems*, vol. 13, 52–8. Cambridge, MA: MIT Press.

Taatgen, N. A., and J. R. Anderson. 2008. ACT-R. In *Constraints in Cognitive Architectures*, ed. R. Sun, 170–85. Cambridge: Cambridge University Press.

Taylor, R. P., B. Spehar, J. A. Wise, C. W. G. Clifford, B. R. Newell, C. M. Hagerhall, T. Purcell, and T. P. Martin. 2005. Perceptual and physiological responses to the visual complexity of Pollock's dripped fractal patterns. *Non-linear Dynamics Psychol Life Sci* 9:89–114.

Thomas, J. C. 1974. An analysis of behavior in the hobbits-orcs problem. *Cogn Psychol* 6:257–69.

Thomas, J. C. 1976. Quantifiers and question-asking. IBM Research Report, RC-5886.

Thomas, J. C. 1983. Psychological issues in the design of database query languages. In *Designing for Human-Computer Communication*, ed. M. E. Sime and M. J. Coombs. London: Academic Press.

Thomas, J. C. 1999. Narrative technology and the new millennium. *Knowl Manag J* 2(9):14–7.

Thomas, J. C., A. Lee, and C. Danis. 2002. "Who Speaks for Wolf?" *IBM Research Report*, RC-22644. Yorktown Heights, NY: IBM Corporation.

Thomas, J. C., and R. Farrell. 2006. HCI Techniques from idea to deployment: A case study for a dynamic learning environment. In *CHI '06 Extended Abstracts On Human Factors In Computing Systems*, April 22–27, 2006, Montréal, Québec, Canada, New York: ACM.

Thomas, J. C., J. L. Fozard, and N. C. Waugh. 1977. Age-related differences in naming latency. *Am J Psychol* 90(30):499–509.

Thomas, J. C., and J. D. Gould. 1975. A psychological study of query by example. In *National Computer Conference Proceedings*, 44, 439–45. New York: AFIPS Press.

Thomas, J. C., W. A. Kellogg, and T. Erickson. 2001. The knowledge management puzzle: Human and social factors in knowledge management. *IBM Syst J* 40(4):863–84. http://www.research.ibm.com/journal/sj40-4.html

Thomas, J. C., and H. Ruben. 1973. Age and mnemonic techniques in paired-associate learning. Presented at the Gerontological Society Meeting, Miami Beach, Florida, November 8, 1973.

Vieweg, S., A. Hughes, K. Starburd, and L. Palen. 2010 Microblogging during two natural hazards events: What twitter may contribute to situational awareness. In *Proceedings of ACM Conference on Human Factors in Computing Systems*. 1079–88. NY: ACM.

Underwood, P. 1983. *Who speaks for Wolf? A Native American learning story*. San Anselmo, CA: Learning Way Company (Now, Tribe of Two Press).

Walston, C. E., and C. P. Felix. 1977. A method of programming measurement and estimation. *IBM Syst J* 16:54–73.

Weinberg, G. M. 1971. *The Psychology of Computer Programming*. New York: Van Nostrand Reinhold.

Werker, J. F., and R. C. Tees. 1984. Cross-language speech perception: evidence of perceptual reorganization during the first year of life. *Infant Behav Dev* 7:49–63.

23 Information Visualization

Stuart Card

CONTENTS

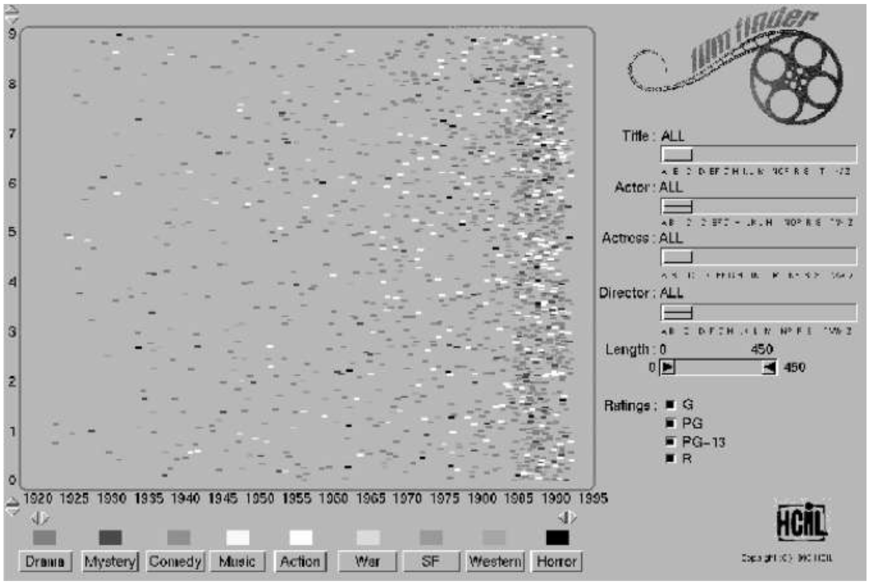
23.1 INTRODUCTION

The working mind is greatly leveraged by interaction with the world outside it. A conversation to share information, a grocery list to aid memory, a pocket calculator to compute square roots all effectively augment a cognitive ability otherwise severely constrained by what is in its limited knowledge, by limited attention, and by limitations on reasoning. But the most profound leverage on cognitive ability is the ability to invent new representations, procedures, or devices that augment cognition far beyond its unaided biological endowment—and bootstrap these into even more potent inventions.

This chapter is about one class of inventions for augmenting cognition, collectively called "information visualization." Other senses could be employed in this pursuit—audition, for example, or a multi-modal combination of senses—the broader topic is really *information perceptualization*; however, in this chapter, we restrict ourselves to visualization. Visualization employs the sense with the most information capacity; recent advances in graphically agile computers have opened opportunities to exploit this capacity, and many visualization techniques have now been developed. A few examples suggest the possibilities.

23.1.1 Example 1: Finding Videos with the FilmFinder

The use of information visualization for finding things is illustrated by the FilmFinder (Ahlberg and Shneiderman 1994a,b). Unlike typical movie-finder systems, the Film Finder is organized not around searching with keywords, but rather around rapid browsing and reacting to collections of films in the database. Figure 23.1 shows a scattergraph of 2000 movies, plotting rated quality of the movie as a function of year when it was released. Color differentiates type of movies—comedy from drama and the like. The display provides an overview, the entire universe of all the movies, and some general features of the collection. It is visually apparent, for example, that a good share of the movies in the collection were released after 1965, but also that there are movies going back as far as the 1920s. Now the viewer "drills down" into the collection by using the sliders in the interface to show only movies with Sean Connery that are between 1 and 4½



Title : ALL

Actor : ALL

Actress : ALL

Director : ALL

Length : 0 450

0 450

Ratings : ■ G ■ PG ■ PG-13 ■ R

FIGURE 23.1 FilmFinder overview scattergraph. (Courtesy University of Maryland.)

hours in length (Figure 23.2). As the sliders are moved, the display zooms in to show about 20 movies. It can be seen that these movies were made between 1960 and 1995, and all have a quality rating higher than 4. Since there is now room on the display, titles of the movies appear. Experimentation with the slider shows that restricting maximum length to 2 hours cuts out few interesting movies. The viewer chooses the highly rated movie, *Murder on the Orient Express* by double-clicking on its marker. Up pop details in a box (Figure 23.3) giving names of other actors in the movie and more information. The viewer is interested in whether two of these actors, Anthony Perkins and Ingrid Bergman, have appeared together in any other movies. The viewer selects their names in the box, and then requests another search (Figure 23.4). The result is a new display of two movies. In addition to the movie the viewer knew about, there is one other movie, a drama entitled *Goodbye, Again*, made around 1960. The viewer is curious about this movie and decides to watch it.

Information visualization has allowed a movie viewer in a matter of seconds to find a movie he or she could not have specified at the outset. To do this, the FilmFinder employed several techniques from information visualization: (a) an *overview* of the collection showing its structure; (b) *dynamic queries*, in which the visualization seems to change instantaneously with control manipulations; (c) *zooming in* by adding restrictions to the set of interest; (d) *details on demand*, in which the user can display temporarily details about an individual object, and (e) *retrieval by example*, in which selected attributes of an individual object are used to specify a new retrieval set.

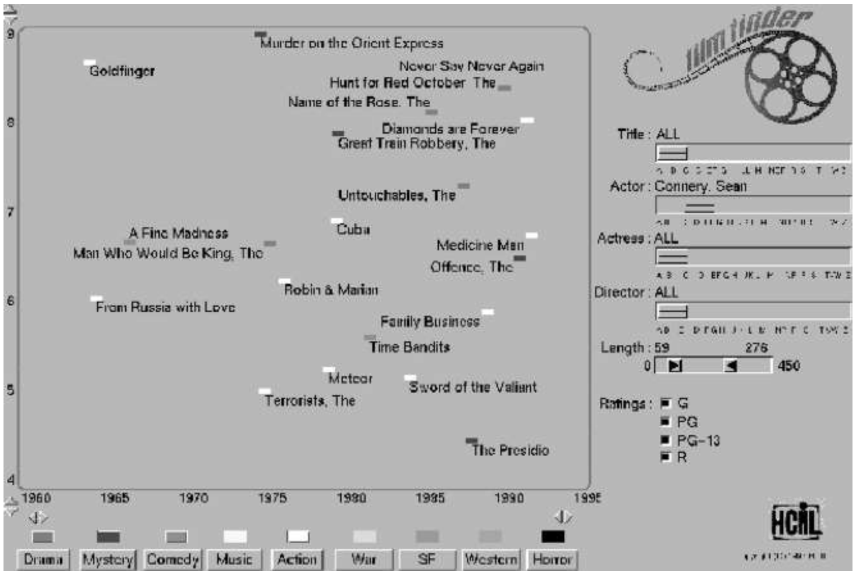
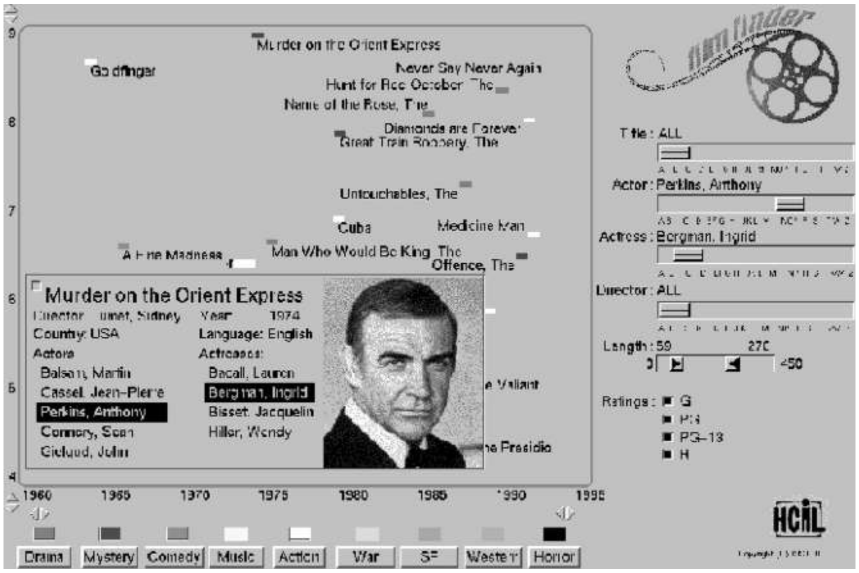


FIGURE 23.2 FilmFinder scattergraph zoom-in. (Courtesy University of Maryland.)


Murder on the Orient Express

Director Lumet, Sidney Year 1974

Country: USA Language: English

Actors
- Balsam, Martin
- Cassel, Jean-Pierre
- Perkins, Anthony
- Connery, Sean
- Gielgud, John

Actresses:
- Bacall, Lauren
- Bergman, Ingrid
- Bisset, Jacquelin
- Hiller, Wendy

FIGURE 23.3 FilmFinder scattergraph zoom-in. (Courtesy University of Maryland.)

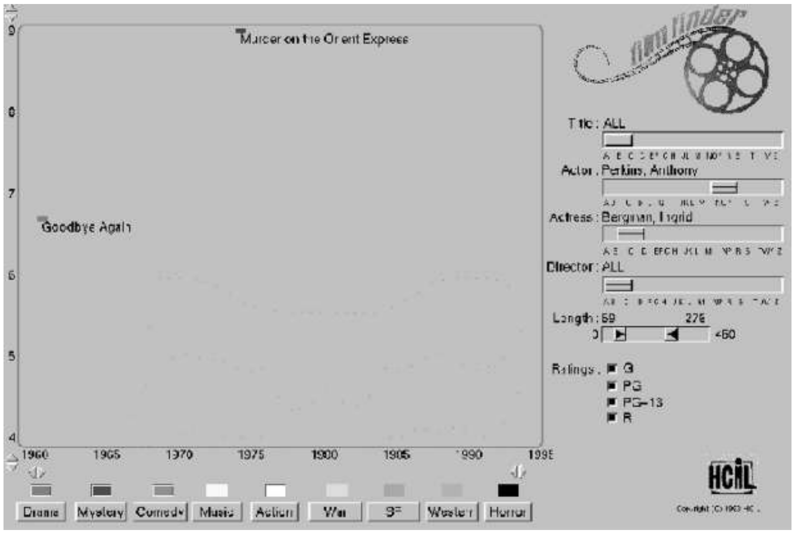
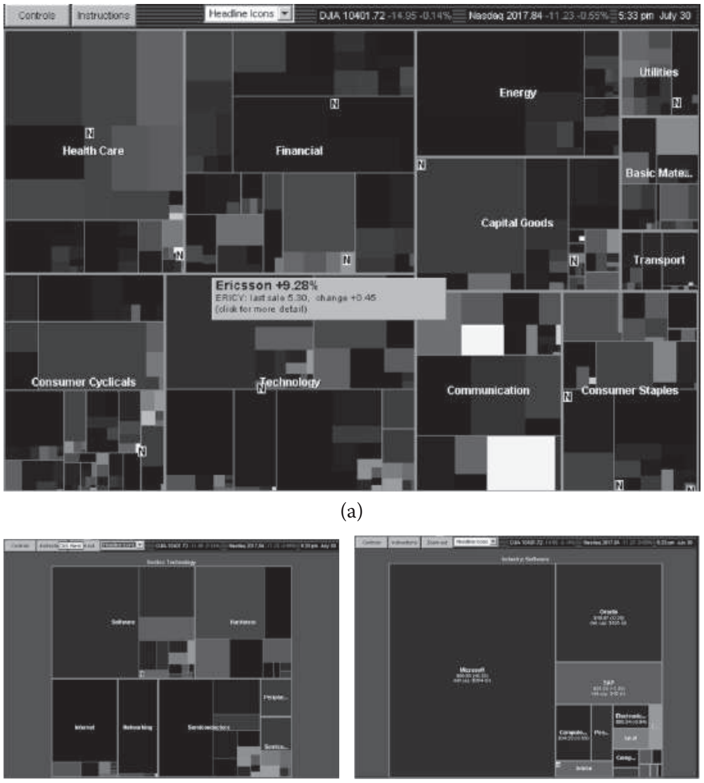


FIGURE 23.4 FilmFinder retrieval by example. (Courtesy University of Maryland.)



(b) (c)

FIGURE 23.5 TreeMap of daily stock prices. (Courtesy SmartMoney.com.)

23.1.2 Example 2: Monitoring Stocks with TreeMaps

Another example of information visualization is the TreeMap visualization on the SmartMoney.com website,* which is shown in Figure 23.5a. Using this visualization, an investor can monitor more than 500 stocks at once, with data updated every 15 minutes. Each colored rectangle in the figure is a company. The size of the rectangle is proportional to its market capitalization. Color of the rectangle shows movement in the stock price. Bright yellow corresponds to about a 6% increase in price, bright blue to about a 6% decrease in price. Each business sector is identified with a label like "Communications." Those items marked with a letter *N* have an associated news item.

In this example, the investor's task is to monitor the day's market and notice interesting developments. In Figure 23.5a, the investor has moved the mouse over one of the bright yellow

* This figure is produced by a program called SiteMap by Xia Lin and Associates. See http://faculty.cis.drexel.edu/sitemap/index.html.

rectangles, and a box identifying it as Erickson, with a +9.28% gain for the day, has popped up together with other information. Clicking on a box gives the investor a pop-up menu for selecting even more detail. The investor can either click to go to World Wide Web links on news or financials, or drill down, for example, to the sector (Figure 23.5b), or down further to individual companies in the software part of the technology sector (Figure 23.5c). The investor is now able to immediately note interesting relationships. The software industry is now larger than the hardware industry, for example, and despite a recent battering at the time of this figure, the Internet industry is also relatively large. Microsoft is larger than all the other companies in its industry combined. Selecting a menu item to look at year-to-date gains (Figure 23.6), the investor immediately notes interesting patterns: Microsoft stock shows substantial gains, whereas Oracle is down; Dell is up, but Compaq is down; Tiny Advanced Micro is up, whereas giant Intel is neutral. Having noticed these relationships, the investor drills down to put up charts or analysts' positions for companies whose gains in themselves, *or in relation to a competitor*, are interesting. For example, the investor is preparing a report on the computer industry for colleagues and notices how AMD is making gains against Intel, or how competition for the Internet is turning into a battle between Microsoft and AOL/Time Warner.

23.1.3 Example 3: Sensemaking with Permutation Matrices

As a final information visualization example, consider the case proposed by Bertin (1981) of a hotel manager who wants to analyze hotel occupancy data (Table 23.1) to increase her return. In order to search for meaningful patterns in her data, she represents it as a permutation matrix (Figure 23.7a). A permutation matrix is a graphic rendition of a cases x variables display. In Figure 23.7a, each cell of Table 23.1 is a small bar of a bar chart. The bars for cells below the mean are white; those above the bar are black. By permuting rows and columns, patterns emerge that lead to making sense of the data.

In Figure 23.7a, the set of months, which form the cases, are repeated to reveal periodic patterns across the end of the cycle. By visually comparing the pairs of rows, one can find rows that are similar. These are reordered and grouped (Figure 23.7b). By this means, it is discovered that there seem to be two patterns of yearly variation. One pattern in Figure 23.7b is semiannual, dividing the year into the cold months of October through April and the warm months of May through September. The other pattern breaks the year into four distinct regions. We have thus found the beginnings of a *schema*—that is, a framework in terms of which we can encode the raw data and describe it in a more compact language. Instead of talking about the events of the year in terms of individual months, we can now talk in terms of two series of periods, the semiannual one, and the four distinct periods. As we do so, there is a *residue* of information not included as part of our descriptive language. Sensemaking proceeds by the *omission and recoding of information into more compact form* (see Resnikoff 1989). This residue of information may be reduced by finding a better or more articulated schema, or it may be left as noise. Beyond finding the basic patterns in the data, the hotel manager wants to make sense of the data relative to a purpose: she wants to increase the occupancy of the hotel. Therefore, she has also permuted general indicators of activity in Figure 23.7b, such as % Occupancy and Length of Stay, to the top of the diagram and put the rows that correlate with these below them. This reveals that Conventions, Businessmen, and Agency Reservations, all of which generally have to do with convention business, are associated with higher occupancy. This insight comes from the match

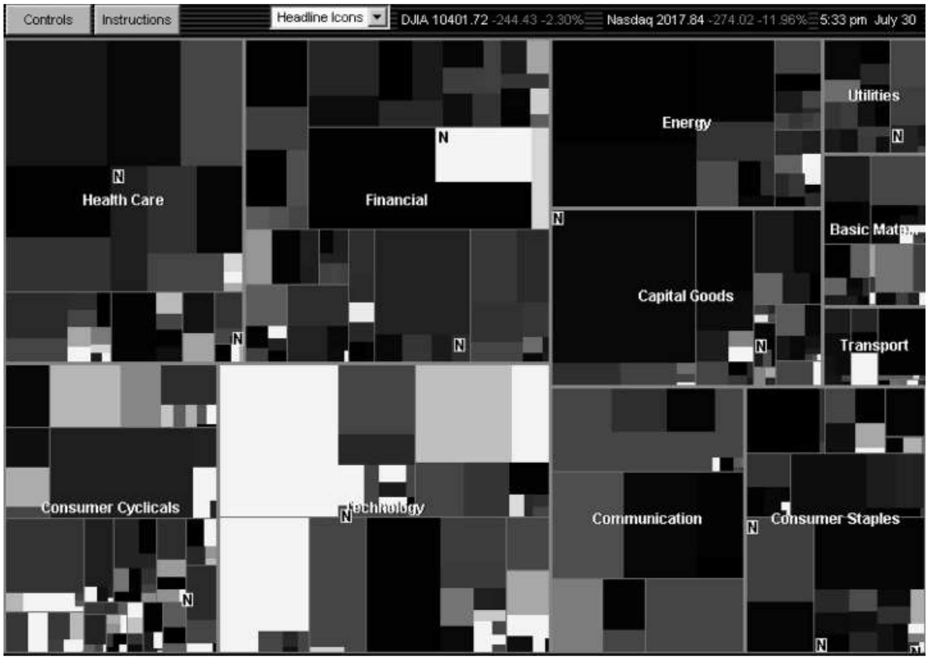


FIGURE 23.6 TreeMap of year-to-date stock prices. (Courtesy SmartMoney.com.)

TABLE 23.1
Data for Hotel Occupancy

ID	VARIABLE	JAN	FEB	MAR	APR	MAY	JUNE	JULY	AUG	SEPT	OCT	NOV	DEC
1	% Female	26	21	26	28	20	20	20	20	20	40	15	40
2	% Local	69	70	77	71	37	36	39	39	55	60	68	72
3	% USA	7	6	3	6	23	14	19	14	9	6	8	8
4	% South America	0	0	0	0	8	6	6	4	2	12	0	0
5	% Europe	20	15	14	15	23	27	22	30	27	19	19	17
6	% M.East/Africa	1	0	0	8	6	4	6	4	2	1	0	1
7	% Asia	3	10	6	0	3	13	8	9	5	2	5	2
8	% Businessmen	78	80	85	86	85	87	70	76	87	85	87	80
9	% Tourists	22	20	15	14	15	13	30	24	13	15	13	20
10	% Direct Reservations	70	70	75	74	69	68	74	75	68	68	64	75
11	% Agency Reservations	20	18	19	17	27	27	19	19	26	27	21	15
12	% Air Crews	10	12	6	9	4	5	7	6	6	5	15	10
13	% Under 20	2	2	4	2	2	1	1	2	2	4	2	5
14	% 20–35	25	27	37	35	25	25	27	28	24	30	24	30
15	% 35–55	48	49	42	48	54	55	53	51	55	46	55	43
16	% Over 55	25	22	17	15	19	19	19	19	19	20	19	22
17	Price of rooms	163	167	166	174	152	155	145	170	157	174	165	156
18	Length of stay	1.7	1.7	1.7	1.91	1.9	2	1.54	1.6	1.73	1.82	1.66	1.44
19	% Occupancy	67	82	70	83	74	77	56	62	90	92	78	55
20	Conventions	0	0	0	1	1	1	0	0	1	1	1	1

Source: Bertin, J. 1981. Graphics and graphic information-processing. Berlin: Walter de Gruyter.

in patterns *internal* to the visualization; it also comes from noting why these variables might correlate as a consequence of factors *external* to the visualization. She also discovers that marked differences exist between the winter and summer guests during the slow periods. In winter, there are more local guests, women, and age differences. In summer, there are more foreign tourists and less variation in age.

This visualization was useful for sensemaking on hotel occupancy data, but it is too complicated to communicate the high points. The hotel manager therefore creates a simplified diagram, Figure 23.7c. By graying some of the bars, the main points are more readily graspable, while still preserving the data relations. A December convention, for example, does not seem to have the effect of the other conventions in bringing in guests. It is shown in gray as residue in the pattern. The hotel manager suggests moving the convention to another month, where it might have more effect on increasing the occupancy of the hotel.

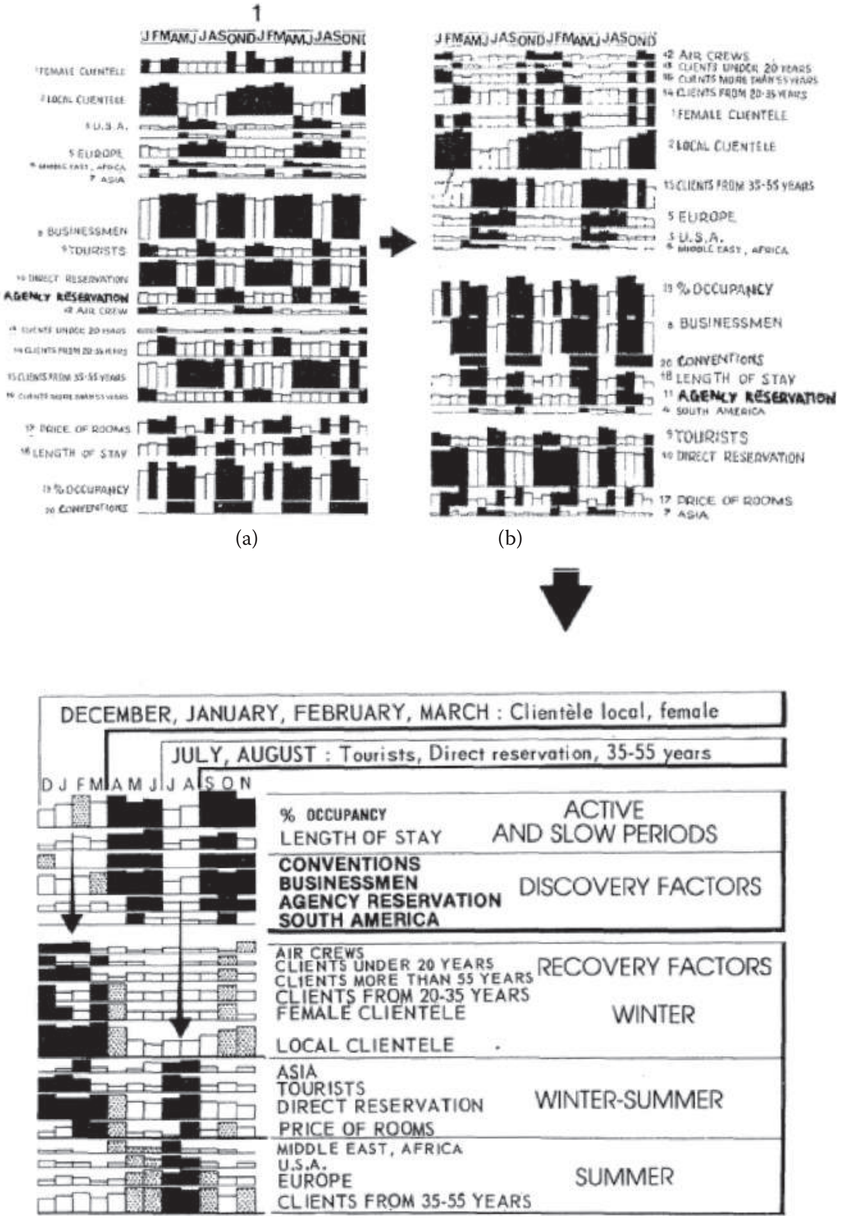
23.1.4 What Is Information Visualization?

The FilmFinder, the TreeMap, and the permutation matrix hotel analysis are all examples of the use of information visualization. We can define information visualization as "the use of computer-supported, interactive, visual representations of abstract data in order to amplify cognition" (Card, Mackinlay, and Shneiderman 1999).

Information visualization needs to be distinguished from related areas: *scientific visualization* is like information visualization, but it is applied to scientific data and typically is physically based. The starting point of a natural geometrical substrate for the data, whether the human body or earth geography, tends to emphasize finding a way to make visible the invisible (say, velocity of air flow) within an existing spatial framework. The chief problem for information visualization, in contrast, is often finding an effective mapping between abstract entities and a spatial representation. Both information visualization and scientific visualization belong to the broader field of *data graphics*, which is the use of abstract, nonrepresentational visual representations to amplify cognition. Data graphics, in turn, is part of *information design*, which concerns itself with external representations for amplifying cognition. At the highest level, we could consider information design a part of *external cognition*, the uses of the external world to accomplish some cognitive process. Characterizing the purpose of information visualization as *amplifying cognition* is purposely broad. Cognition can be the process of writing a scientific paper or shopping on the Internet for a cell phone. Generally, it refers to the intellectual processes in which information is obtained, transformed, stored, retrieved, and used. All of these can be advanced generally by means of external cognition, and in particular by means of information visualization.

23.1.5 Why Does Visualization Work?

Visualization aids cognition not because of some mystical superiority of pictures over other forms of thought and communication, but rather because visualization helps the user by making the world outside the mind a resource for thought in



(a)

(b)

(c)

FIGURE 23.7 Permutation matrix representation of hotel data (Berlin 1977/1981). (a) Initial matrix of variables. (b) Permuted matrix to group like patterns together.

specific ways. We list six groups of these in Table 23.2 (Card, Mackinlay, and Shneiderman 1999): Visualization amplifies cognition by (1) increasing the memory and processing resources available to the users, (2) reducing search for information, (3) using visual representations to enhance the detection of patterns, (4) enabling perceptual inference operations, (5) using perceptual attention mechanisms for monitoring, and (6) by encoding information in a manipulable medium. The FilmFinder, for example, allows the representation of a large amount of data in a small space in a way that allows patterns to be perceived visually in the data. Most important, the method of instantly responding in the display to the dynamic movement of the sliders allowed users to rapidly explore the multidimensional space of films. The TreeMap of the stock market allows monitoring and exploration of many equities. Again, much data is represented in little space. In this case, the display manages the user's attention, drawing it to those equities with unusually large changes, and supplying the means to drill down into the data to understand why these movements may be happening. In the hotel management case, the visual representation makes it easier to notice similarities of behavior in a multidimensional attribute space, then to cluster and represent these. The final product is a compact (and simplified) representation of the original data that supports a set of forward decisions. In all of these cases, visualization allows the user to (a) examine a large amount of information, (b) keep an overview of the whole while pursuing details, (c) keep track of (by using the display as an

TABLE 23.2
How Information Visualization Amplifies Cognition

1. Increased Resources	
High-bandwidth hierarchical interaction	Human moving gaze system partitions limited channel capacity so that it combines high spatial resolution and wide aperture in sensing the visual environments (Larkin and Simon 1987).
Parallel perceptual processing	Some attributes of visualizations can be processed in parallel compared to text, which is serial.
Offload work from cognitive to perceptual system	Some cognitive inferences done symbolically can be recoded into inferences done with simple perceptual operations (Larkin and Simon 1987).
Expanded working memory	Visualizations can expand the working memory available for solving a problem (Norman 1993).
Expanded storage of information	Visualizations can be used to store massive amounts of information in a quickly accessible form (e.g., maps).
2. Reduced Search	
Locality of processing	Visualizations group information used together reducing search (Larkin and Simon 1987).
High data density	Visualizations can often represent a large amount of data in a small space (Tufte 1983).
Spatially-indexed addressing	By grouping data about an object, visualizations can avoid symbolic labels (Larkin and Simon 1987).
3. Enhanced Recognition of Patterns	
Recognition instead of recall	Recognizing information generated by a visualization is easier than recalling that information by the user.
Abstraction and aggregation	Visualizations simplify and organize information, supplying higher centers with aggregated forms of information through abstraction and selective omission (Card, Robertson, and Mackinlay 1991; Resnikoff 1989).
Visual schemata for organization	Visually organizing data by structural relationships (e.g., by time) enhances patterns.
Value, relationship, trend	Visualizations can be constructed to enhance patterns at all three levels (Bertin 1967/1983).
4. Perceptual Inference	
Visual representations make some problems obvious	Visualizations can support a large number of perceptual inferences that are very easy for humans (Larkin and Simon 1987).
Graphical computations	Visualizations can enable complex specialized graphical computations (Hutchins 1996).
5. Perceptual Monitoring	Visualizations can allow for the monitoring of a large number of potential events if the display is organized so that these stand out by appearance or motion.
6. Manipulable Medium	Unlike static diagrams, visualizations can allow exploration of a space of parameter values and can amplify user operations.

Source: Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.

external working memory) many things, and (d) produce an abstract representation of a situation through the omission and recoding of information.

23.1.6 Historical Origins

Drawn visual representations have a long history. Maps go back millennia. Diagrams were an important part of Euclid's books on geometry. Science, from earliest times, used diagrams to (a) record observations, (b) induct relationships, (c) explicate methodology of experiments, and (d) classify and conceptualize phenomena (for a discussion, see Robin 1992). For example, Figure 23.8 is a hand-drawn illustration in Newton's first scientific publication, illustrating how white light is really composed of many colors. Sunlight enters from the window at right and is refracted into many colors by a prism. One of these colors can be selected (by an aperture in a screen) and further refracted by another prism, but the light stays the same color, showing that it has already been reduced to its elementary components. As in Newton's illustration, early scientific and mathematical diagrams generally had a spatial, physical basis and were used to reveal the hidden, underlying order in that world.

Surprisingly, diagrams of abstract, nonphysical information are apparently rather recent. Tufte (1983) dates abstract diagrams to Playfair (1786) in the 18th century. Figure 23.9 is one of Playfair's earliest diagrams. The purpose was to convince readers that English imports were catching up with exports. Starting with Playfair, the classical methods of plotting data were developed—graphs, bar charts, and the rest.

Recent advances in the visual representation of abstract information derive from several strands that became intertwined. In 1967, Bertin (1967/1983, 1977/1981), a French cartographer, published his theory of *The Semiology of Graphics*. This theory identified the basic elements of diagrams and their combination. Tufte (1983, 1990, 1997), from the fields of visual design and data graphics, published a series of seminal books that set forth principles for the design of data graphics and emphasized maximizing the density of useful information. Both Bertin's and Tufte's theories became well known and influential. Meanwhile, within statistics, Tukey (1977) began a movement on exploratory data analysis. His emphasis was not on the quality of graphical presentation, but on the use of pictures to give rapid, statistical insight into data relations. For example, "box and whisker plots" allowed an analyst to get a rapid characterization of data distributions. Cleveland and McGill (1988) wrote an influential book, *Dynamic Graphics for Statistics*, explicating new visualizations of data with particular emphasis on the visualization of multidimensional data.

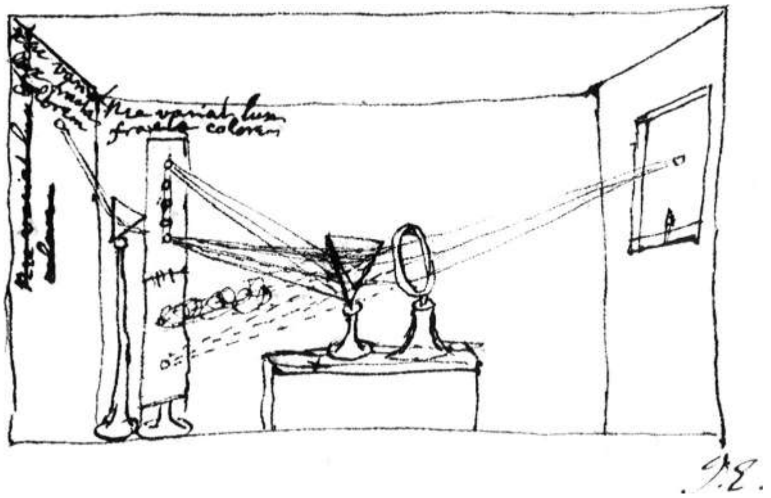


FIGURE 23.8 Newton's optics illustration. (Data from Robin, H. 1992. *The Scientific Image: From Cave to Computer*. New York: H. N. Abrams, Inc.)

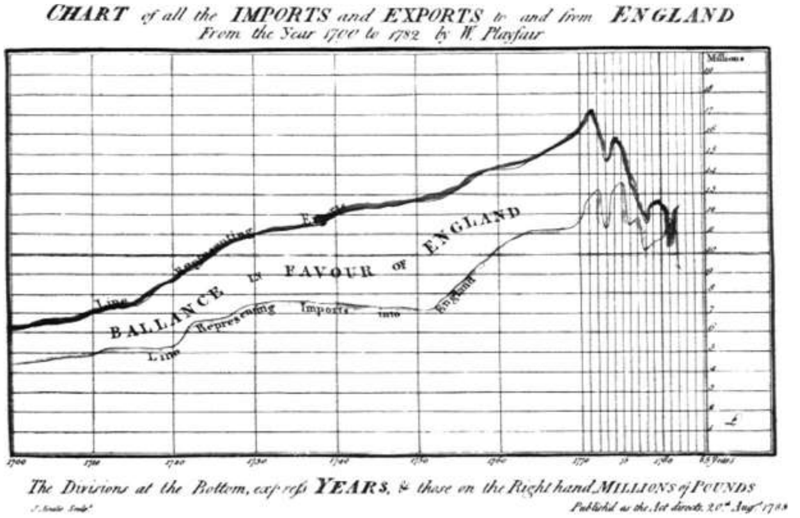


FIGURE 23.9 Playfair's charts of English imports and exports. (Data from Tufte, E. R. 1983. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.)

In 1985, NSF launched an initiative on *scientific visualization* (McCormick and DeFanti 1987). The purpose of this initiative was to use advances in computer graphics to create a new class of analytical instruments for scientific analysis, especially as a tool for comprehending large, newly produced datasets in the geophysical and biological sciences. Meanwhile, the computer graphics and artificial intelligence communities were interested in the automatic design of visual presentations of data. Mackinlay's (1986a,b) thesis APT formalized Bertin's design theory, added psychophysical data, and used these to build a system for automatically generating diagrams of data, tailored for some purpose. Roth and Mattis (1990) built a system to do more complex visualizations, such as some of those from Tufte. Casner (1991) added a representation of tasks. This community was interested not so much in the quality of the graphics as in the automation of the match between data characteristics, presentational purpose, and graphical presentation. Finally, the user interface community saw advances in graphics hardware opening the possibility of a new generation of user interfaces. The first use of the term "information visualization" was probably in Robertson, Card, and Mackinlay (1989). Early studies in this community focused on user interaction with large amounts of information: Feiner and Beshers (1990) presented a method, worlds within worlds, for showing six-dimensional financial data in an immersive virtual reality. Shneiderman (1992) developed a technique called "dynamic queries" for interactively selecting subsets of data items and TreeMaps, a space-filling representation for trees. Robertson, Card, and Mackinlay (1993) presented ways of using animation and distortion to interact with large data sets in a system called the Information Visualizer, which used *focus + context* displays to nonuniformly present large amounts of information. The emphasis for these studies was on the means for cognitive amplification, rather than on the quality of the graphics presentations.

The remainder of this chapter will concentrate on the techniques that have been developed for mapping abstract information to interactive visual form to aid some intellectual task. The perceptual foundations of this effort are beyond the scope of this chapter, but are covered in Ware (2000). Further details on information visualization techniques are addressed in a text by Spence (2000). The classic papers in information visualization are collected in Card, Mackinlay, and Shneiderman (1999).

23.2 VISUALIZATION REFERENCE MODEL

23.2.1 Mapping Data to Visual Form

Despite their seeming variability, information visualizations can be systematically analyzed. Visualizations can be thought of as adjustable mappings from data to visual form to the human perceiver. In fact, we can draw a simple Visualization Reference Model of these mappings (Figure 23.10). Arrows follow from *raw data* (data in some idiosyncratic format) on the left, though a set of *data transformations* into *data tables* (canonical descriptions of data in a variables x cases format extended to include metadata). The most important mapping is the arrow from data tables to *visual structures* (structures that combine values an available vocabulary of visual elements—spatial substrates, marks, and graphical properties). Visual structures can be further transformed by *view transformations*, such as visual distortion or 3D viewing angle, until it finally forms a *view* that can be perceived by human users. Thus, raw data might start out as text represented as indexed strings or arrays. These might be transformed into *document vectors*, normalized vectors in a space with dimensionality as large as the number of words. Document vectors, in turn, might be reduced by multidimensional scaling to create the analytic abstraction to be visualized, expressed as a data table of x, y, z coordinates that could be displayed. These coordinates might be transformed into a visual structure—that is, a surface on an information landscape—which is then viewed at a certain angle.

Similar final effects can be achieved by transformations at different places in the model: When a point is deleted from the visualization, has the point been deleted from the dataset? Or is it still in the data merely not displayed? Chi and Riedl (1998) called this the *view-value distinction*, and it is an example of just one issue where identifying the locus of a transformation using the Visualization Reference Model helps to avoid confusion.

Information visualization is not just about the creation of visual images, but also the interaction with those images in the service of some problem. In the Visualization Reference Model, another set of arrows flow back from the human at the right into the transformations themselves, indicating the adjustment of these transformations by user-operated controls. It is the rapid reciprocal reaction between the generation of images by machine and the selection and parametric adjustment of those images, giving rise to new images that gives rise to the attractive power of interactive information visualization.

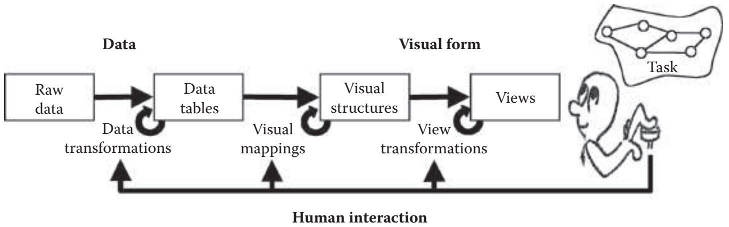
23.2.2 Data Structures

It is convenient to express data tables as tables of objects and their attributes, as in Table 23.3. For example, in the

TABLE 23.3
Data Table about Films

FilmID	**230**	**105**	**540**	**. . .**
Title	Goldfinger	Ben Hur	Ben Hur	. . .
Director	Hamilton	Wyler	Niblo	. . .
Actor	Connery	Heston	Novarro	. . .
Actress	Blackman	Harareet	McAvoy	. . .
Year	1964	1959	1926	. . .
Length	112	212	133	. . .
Popularity	7.7	8.2	7.4	
Rating	PG	G	G	. . .
FilmType	Action	Action	Drama	. . .

Source: Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.



Raw data: idiosyncratic formats
Data tables: relations (cases by variables) + meta-data
Visual structures: spatial substrates + marks + graphical properties
Views: graphical parameters (position, scaling, clipping, . . .)

FIGURE 23.10 Reference model for visualization. Visualization can be described as the mapping of data to visual form that supports human interaction in a workplace for visual sense making. (Data from Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.)

FilmFinder, the basic objects (or "cases") are films. Each film is associated with a number of attributes or variables, such as title, stars, year of release, genre type, and so forth. The vertical double black line in the table separates data in the table to the left of the line from the metadata, expressed as variable names, to the left of the line. The horizontal black line across the table separates input variables from output variables—that is, the table can be thought of as a function.

$$f(\text{input variables}) = \text{output variables}$$

So,

$$\mathit{Year}\ (\mathit{FilmID} = 105) = 1959$$

Variables imply a scale of measurement, and it is important to keep these straight. The most important to distinguish are

N = *Nominal (are only = or ≠ to other values)*
O = *Ordinal* (obeys a < relation)
Q = *Quantitative* (can do arithmetic on them)

A nominal variable N is an unordered set, such as film titles {Goldfinger, Ben Hur, Star Wars}. An ordinal variable O is a tuple (ordered set), such as film ratings ⟨G, PG, PG-13, R⟩. A quantitative variable Q is a numeric range, such as film length [0, 360].

In addition to the three basic types of variables, subtypes represent important properties of the world associated with specialized visual conventions. We sometimes distinguish the subtype *Quantitative Spatial* (Q_s) for intrinsically spatial variables common in scientific visualization and the subtype *Quantitative Geographical* (Q_g) for spatial variables that are specifically geophysical coordinates. Other important subtypes are similarity metrics *Quantitative Similarity* (Q_m), and the temporal variables *Quantitative Time* (Q_t) and *Ordinal Time* (O_t). We can also distinguish Interval Scales (I) (like Quantitative Scales, but since there is not a natural zero point, it is not meaningful to take ratios). An example would be dates. It is meaningful to subtract two dates (June 5, 2002 – June 3, 2002 = 2 days), but it does not make sense to divide them (June 5, 2002 ÷ June 23, 2002 = Undefined). Finally, we can define an *Unstructured* Scale (U), whose only value is present or absent (e.g., an error flag). The scales are summarized in Table 23.4.

Scale types can be altered by transformations, and this practice is sometimes convenient. For example, quantitative variables can be mapped by data transformations into ordinal variables

$$Q \rightarrow O$$

by dividing them into ranges. For example, film lengths [0, 360] minutes (type Q) can be broken into the ranges (type O),

$$[0, 360] \text{ minutes} \rightarrow \langle \text{SHORT, MEDIUM, LONG} \rangle$$

This common transformation is called "classing," because it maps values onto classes of values. It creates an accessible summary of the data, although it loses information. In the other direction, nominal variables can be transformed to ordinal values

$$N \rightarrow O$$

based on their name. For example, film titles {GOLDFINGER, BEN HUR, STAR WARS} can be sorted lexicographically

$$\{\text{GOLDFINGER, BEN HUR, STAR WARS}\} \rightarrow$$
$$\langle \text{BEN HUR, GOLDFINGER, STAR WARS} \rangle$$

Strictly speaking, we have not transformed their values, but in many uses (e.g., building alphabetically arranged dictionaries of words or sliders in the FilmFinder), we can act as if we had.

TABLE 23.4
Classes of Data and Visual Elements

	Data Classes		**Visual Classes**	
Class	**Description**	**Example**	**Description**	**Example**
U	*Unstructured* (can only distinguish presence or absence)	ErrorFlag	*Unstructured* (no axis, indicated merely whether something is present or absent)	Dot
N	*Nominal* (can only distinguish whether two values are equal)	{Goldfinger, Ben Hur, Star Wars}	*Nominal Grid* (a region is divided into subregions, in which something can be present or absent)	Colored circle
O	*Ordinal* (can distinguish whether one value is less or greater but not difference or ratio)	⟨Small, Medium, Large⟩	*Ordinal Grid* (order of the subregions is meaningful)	Alpha slider
I	*Interval* (can do subtraction on values, but no natural zero and can't compute ratios)	[10 Dec. 1978–4 Jun. 1982]	*Interval Grid* (region has a metric but no distinguished origin)	Year axis
Q	*Quantitative* (can do arithmetic on values)	[0–100] kg	*Quantitative Grid* (a region has a metric)	Time slider
Q_s	*—Spatial variables*	[0–20] m	*—Spatial grid*	
Q_m	*—Similarity*	[0–1]	*—Similarity space*	
Q_g	*—Geographical coord.*	[30°N–50°N]Lat.	*—Geographical coord.*	
Q_t	*—Time variable*	[10–20] µsec	*—Time grid*	

TABLE 23.5
Data Table with Meta-Data Describing the Types of the Variables

FilmID	***N***	**230**	**105**	**. . .**
Title	*N*	Goldfinger	Ben Hur	. . .
Director	*N*	Hamilton	Wyler	. . .
Actor	*N*	Connery	Heston	. . .
Actress	*N*	Blackman	Harareet	. . .
Year	Q_t	1964	1959	. . .
Length	*Q*	112	212	. . .
Popularity	*Q*	7.7	8.2	
Rating	*O*	PG	G	. . .
FilmType	*N*	Action	Action	. . .

Source: Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.

Variable scale types form an important class of metadata that, as we shall see, is important for proper information visualization. We can add scale type to our Data Table in Table 23.3 together with cardinality or range of the data to give us essentially a codebook of variables as in Table 23.5.

23.2.3 Visual Structures

Information visualization maps data relations into visual form. At first, it might seem that a hopelessly open set of visual forms can result. Careful reflection, however, reveals what every artist knows: that visual form is subject to strong constraints. Visual form that reflects the systematic mapping of data relations onto visual form, as in information visualization or data graphics, is subject to even more constraints. It is a genuinely surprising fact, therefore, that most information visualization involves the mapping data relations onto only a half dozen components of visual encoding:

1. *Spatial substrate*
2. *Marks*
3. *Connection*
4. *Enclosure*
5. *Retinal properties*
6. *Temporal encoding*

Of these mappings, the most powerful is how data are mapped onto the spatial substrate—that is, how data are mapped into spatial position. In fact, one might say that the design of an information visualization consists first of deciding which variables are going to get the spatial mappings, and then how the rest of the variables are going to make do with the coding mappings that are left.

23.2.3.1 Spatial Substrate

As we have just said, the most important choice in designing an information visualization is which variables are going to map onto spatial position. This decision gives importance to spatially encoded variables at the expense of variables encoded using other mappings. Space is perceptually dominant (MacEachren 1995); it is good for discriminating values and picking out patterns. It is easier, for example, to identify the difference between a sine and a tangent curve when encoded as a sequence of spatial positions than as a sequence of color hues.

Empty space itself, as a container, can be treated as if it had metric structure. Just as we classified variables according to their scale type, we can think of the properties of space in terms of the scale type of an axis of space (cf. Engelhardt et al. 1996). Axis scale types correspond to the variable scale types (see Table 23.4). The most important axes are

U = Unstructured (no axis, indicated merely whether something is present or absent)
N = Nominal Grid (a region is divided into subregions, in which something can be present or absent)
O = Ordinal Grid (the ordering of these subregions is meaningful), and
Q = Quantitative Grid (a region has a metric)

Besides these, it is convenient to make additional distinctions for frequently used subtypes, such as Spatial axes (Qs).

Axes can be linear or radial; essentially, they can involve any of the various coordinate systems for describing space. Axes are an important building block for developing Visual Structures. Based on the Data Table for the FilmFinder in Table 23.5, we represent the scatterplot as composed of two orthogonal quantitative axes:

$$Year \rightarrow Q_x$$
$$Popularity \rightarrow Q_y$$

The notation states that the Year variable is mapped to a quantitative X-axis and the Popularity variable is mapped to a quantitative Y-axis. Other axes are used for the FilmFinder query widgets. For example, an ordinal axis is used in the radio buttons for film ratings,

$$Ratings \rightarrow O_y$$

and a nominal axis is used in the radio buttons for film type,

$$FilmType \rightarrow N_x$$

23.2.3.2 Marks

Marks are the visible things that occur in space. There are four elementary types of marks (Figure 23.11):

1. P = Points (0D)
2. L = Lines (1D)
3. A = Areas (2D)
4. V = Volumes (3D)

Area marks include surfaces in three dimensions, as well as 2D-bounded regions.

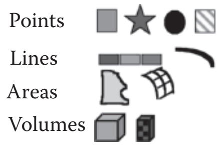


FIGURE 23.11 Types of marks.

Unlike their mathematical counterpart, point and line marks actually take up space (otherwise, they would be invisible) and may have properties such as shape.

23.2.3.3 Connection and Enclosure

Point marks and line marks can be used to signify other sorts of topological structure: graphs and trees. These allow showing relations among objects without the geometrical constraints implicit in mapping variables onto spatial axes. Instead, we draw explicit lines. Hierarchies and other relationships can also be encoded using enclosure. Enclosing lines can be drawn around subsets of items. Enclosure can be used for trees, contour maps, and Venn diagrams.

23.2.3.4 Retinal Properties

Other graphical properties were called retinal properties by Bertin (1967/1983), because the retina of the eye is sensitive to them independent of position. For example, the FilmFinder in Figure 23.1 uses color to encode information in the scatterplot:

$$FilmID(FilmType) \rightarrow P(Color)$$

This notation says that the FilmType attribute for any FilmID case is visually mapped onto the color of a point.

Figure 23.12 shows Bertin's six "retinal variables" separated into spatial properties and object properties according to which area of the brain they are believed to be processed (Kosslyn 1994). They are sorted according to whether the property is good for expressing the extent of a scale (has a natural zero point), or whether its principal use is for differentiating marks (Bertin 1977/1981). Spatial position, discussed earlier as basic visual substrate, is shown in the position it would occupy in this classification.

Other graphical properties have also been proposed for encoding information. MacEachren (1995) has proposed (a) crispness (the inverse of the amount of distance used to blend two areas or a line into an area), (b) resolution (grain with raster or vector data will be displayed), (c) transparency, and (d) arrangement (e.g., different ways of configuring dots). He further proposed dividing color into (a) value (essentially, the gray level of Figure 23.12), (b) hue, and (c) saturation. Graphical properties from the perception literature that can support preattentive processing have been suggested candidates for coding variables such as curvature, lighting direction, or direction of motion (see Healey, Booth, and Enns 1995). All of these suggestions require further research.

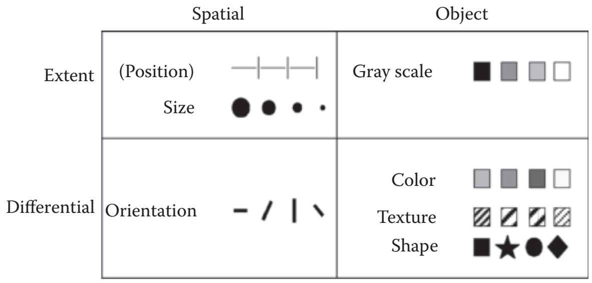


FIGURE 23.12 Retinal properties. The six retinal properties can be grouped by whether they form a scale with a natural zero point (extend) and whether they deal with spatial distance or orientation (spatial). (Data from Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.)

23.2.3.5 Temporal Encoding

Visual Structures can also temporally encode information; human perception is very sensitive to changes in mark position and the mark's retinal properties. We need to distinguish between temporal data variables to be visualized:

$$\mathrm{Q}_t \rightarrow \textit{some visual representation}$$

and animation, that is, mapping a variable into time,

$$\textit{some variable} \rightarrow \textit{Time}$$

Time as animation could encode any type of data (whether it would be an effective encoding is another matter). Time as animation, of course, can be used to visualize time as data.

$$\mathrm{Q}_t \rightarrow \textit{Time}$$

This is natural, but not always the most effective encoding. Mapping time data into space allows comparisons between two points in time. For example, if we map time and a function of time into space (e.g., time and accumulated rainfall),

$$Q_t \rightarrow Q_x \textit{ [make time be the X-axis]}$$
$$f(Q_t) \rightarrow Q_y \textit{ [make accumulated rainfall be the Y-axis]},$$

then we can directly experience rates as visual linear slope, and we can experience changes in rates as curves. This encoding of time into space for display allows us to make much more precise judgments about rates than would be possible from encoding time as time. Another use of time as animation is similar to the unstructured axes of space. Animation can be used to enhance the ability of the user to keep track changes of view or visualization. If the user clicks on some structure, causing it to enlarge and other structures to become smaller, animation can effectively convey the change and the identity of objects across the change, whereas simply viewing the two end states is confusing. Another use is to enhance a visual effect. Rotating a complicated object, for example, will induce 3D effects (hence, allow better reading of some visual mappings).

23.2.4 Expressiveness and Effectiveness

Visual mappings transform Data Tables into Visual Structure and then into a visual image. This image is not just an arbitrary image. It is an image that has a particular meaning it must express. That meaning is the data relation of which it is the visual transformation. We can think of the image as a sentence in a visual language (Mackinlay 1986b) that expresses the relations in the Data Table. To be a good information visualization, the mappings must satisfy some constraints. The first constraint is that the mapping must be expressive. A visualization is said to be *expressive* if and only if it encodes all the data relations intended and no other data relations. The first part of expressiveness turns out to be easier than the second. Suppose we plot FilmType against Year using the data-to-visual mapping in Figure 23.13. The problem of this mapping is that the nominal movie rating data are expressed by a quantitative axis. That is, we have tried to map:

$$FilmType(N) \rightarrow Position(Q)$$

In so doing, we have visually expressed all the data relation, but the visualization also implies relationships that do not exist. For example, the 1959 version of *Ben Hur* does not have a film type that is five times greater than the 1926 version of *Ben Hur,* as implied in the figure. Wisely, the authors of the FilmFinder chose the mapping:

$$FilmType(N) \rightarrow Color(N)$$

Of course, there are circumstances in which color could be read as ordinal, or even possibly quantitative, but the miscellaneous order of the buttons in Figure 23.1 discourages such an interpretation and the relatively low effectiveness of color for this purpose in Table 23.7 also discourages this interpretation.

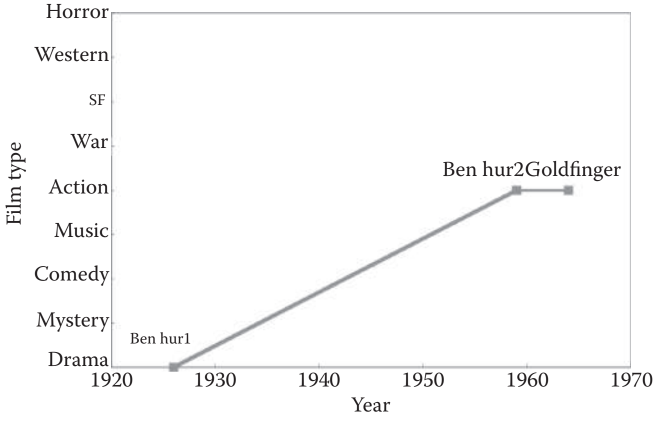


FIGURE 23.13 Mapping from data to visual form that violates expressiveness criterion.

Table 23.6 shows the mappings chosen by authors of the FilmFinder. The figure shows the Data Table's metadata and data and how they are mapped onto the Visual Structure. Note that the nominal data of the PG ratings is mapped onto a nominal visualization technique (colors). Note also, that names of directors and stars (nominal variables) are raised to ordinal variables (through alphabetization), and then mapped onto an ordinal axis. This is, of course, a common way to handle searching among a large number of nominal items.

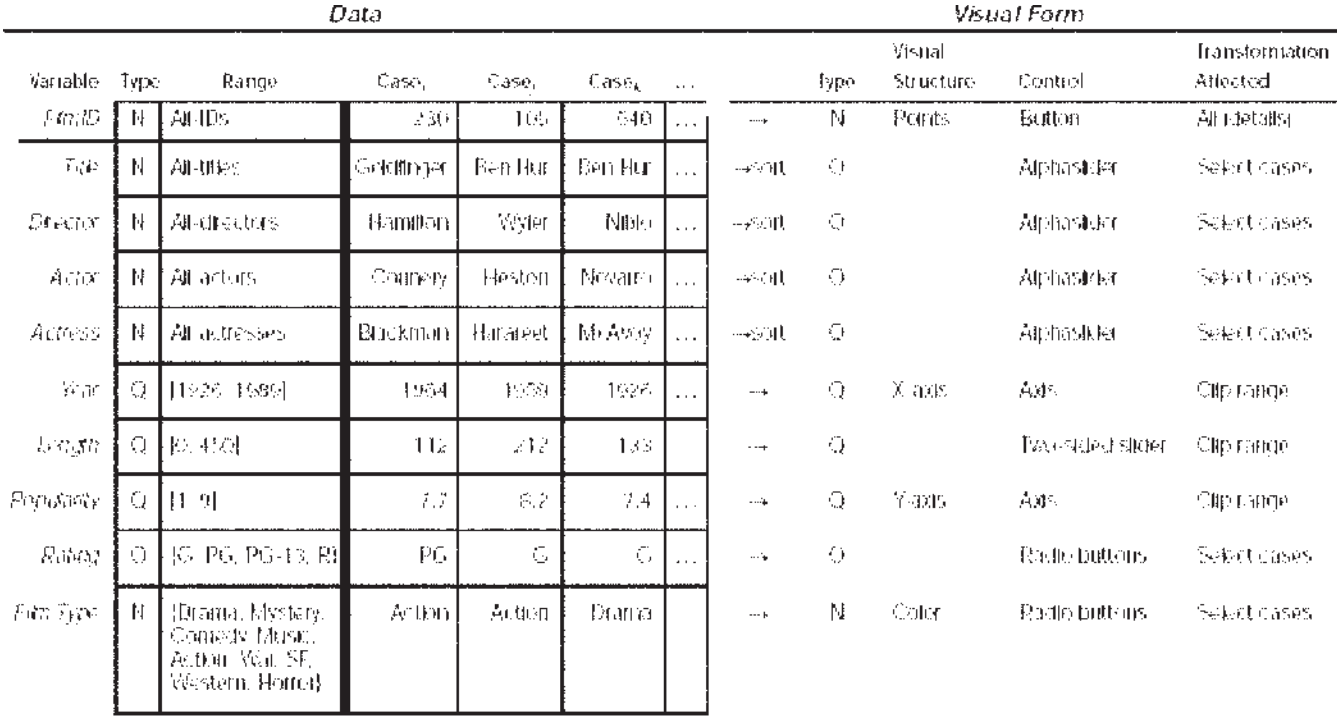
Some properties are more effective than others for encoding information. Position is by far the most effective all-around representation. Many properties are more effective for some types of data than for others. Table 23.7 gives an approximate evaluation for the relative effectiveness of some encoding techniques based on (MacEachren 1995). We note that spatial position is effective for all scale types of data. Shape, on the other hand, is only effective for nominal data. Gray scale is most effective for ordinal data. Such a chart can suggest representations to a visualization designer.

23.2.5 Taxonomy of Information Visualizations

We have shown that the properties of data and visual representation generally constrain the set of mappings that form the basis for information visualizations. Taken together, these constraints form the basis of a taxonomy of information visualizations. Such a taxonomy is given in Table 23.8. Visualizations are grouped into four categories. First are *Simple Visual Structures*, the static mapping of data onto multiple spatial dimensions, trees, or networks plus retinal variables, depicted in Figure 23.10. Here it is worth distinguishing two cases. There is a perceptual barrier at three (or, in special cases, four) variables, a limit of the amount of data that can be perceived as an immediate whole. Bertin (1977, 1981) called this elementary unit of visual data perception the "image." Although this limit has not been definitively established in information visualization by empirical research, there must be a limit somewhere or else people could simultaneously comprehend a thousand variables. We therefore divide visualizations into those that can be comprehended in an elementary perceptual grasp (three, or in special cases, four variables)—let us call these *direct reading visualization*s—and those more complex than that barrier—which we call *articulated reading visualizations*, in which multiple actions are required.

Beyond the perceptual barrier, direct composition of data relationships in terms of 1, 2, or 3 spatial dimensions plus remaining retinal variables is still possible, but rapidly diminishes in effectiveness. In fact, the main problem of information visualization as a discipline can be seen as devising techniques for accelerating the comprehension of these more complex *n*-variable data relations. Several classes of techniques for *n*-variable visualization, which we call *Composed Visual Structures*, are based on composing Simple Visual Structures together by reusing their spatial axes. A third class of Visual Structures—*Interactive Visual Structures*—comes from using the rapid interaction capabilities of the computer. These visualizations invoke

TABLE 23.6
Meta-Data and Mappings of Data onto Visual Structure in the FilmFinder



Data							*Visual Form*				
Variable	Type	Range	Case$_i$	Case$_j$	Case$_k$	...		Type	Visual Structure	Control	Transformation Affected
FilmID	N	All IDs	230	105	540	...	→	N	Points	Button	All details
Title	N	All titles	Goldfinger	Ben Hur	Ben Hur	...	→sort	O		Alphaslider	Select cases
Director	N	All directors	Hamilton	Wyler	Niblo	...	→sort	O		Alphaslider	Select cases
Actor	N	All actors	Connery	Heston	Novarro	...	→sort	O		Alphaslider	Select cases
Actress	N	All actresses	Blackman	Harareet	McAvoy	...	→sort	O		Alphaslider	Select cases
Year	Q	[1926, 1989]	1964	1959	1926	...	→	Q	X-axis	Axis	Clip range
Length	Q	[0, 450]	112	212	133		→	Q		Two-sided slider	Clip range
Popularity	Q	[1, 9]	7.7	8.2	7.4	...	→	Q	Y-axis	Axis	Clip range
Rating	O	{G, PG, PG-13, R}	PG	G	G	...	→	O		Radio buttons	Select cases
Film Type	N	{Drama, Mystery, Comedy, Music, Action, War, SF, Western, Horror}	Action	Action	Drama		→	N	Color	Radio buttons	Select cases

Source: Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.

TABLE 23.7
Relative Effectiveness of Position and Retinal Encodings

	Spatial	**Q**	**O**	**N**	**Object**	**Q**	**O**	**N**
Extent	(Position)	●	●	●	Gray scale	◐	●	○
	Size	●	●	●	Color	◐	◐	●
Differential	Orientation	◐	●	●	Texture	◐	◐	●
					Shape	○	○	●

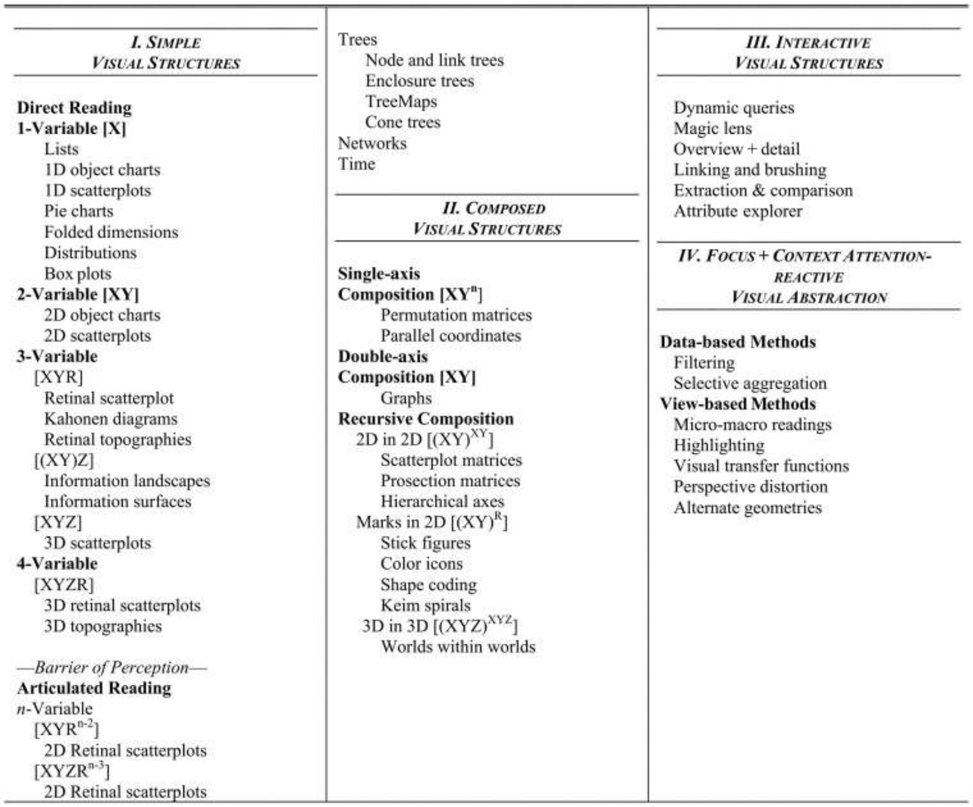
Source: Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)*, 181–8.

the parameter-controlling arrows of Figure 23.10. Finally, a fourth class of visualizations—*Attention-Reactive Visual Structures*—comes from interactive displays where the system reacts to user actions by changing the display, even anticipating new displays, to lower the cost of information access and sensemaking to the user. To summarize,

I. Simple Visual Structures
Direct Reading
Articulated Reading
II. Composed Visual Structures
Single-Axis Composition
Double-Axis Composition
Recursive Composition
III. Interactive Visual Structure
IV. Attention-Reactive Visual Structure

These classes of techniques may be combined to produce visualizations that are more complex. To help us keep track of the variable mapping into visual structure, we will use a simple shorthand notation for listing the element of the Visual Structure that the Data Table has mapped into. We will write, for example, [XYR^2] to note that variables map onto the X-axis, the Y-axis, and two retinal encodings. [OX] will indicate that the variables map onto one spatial axis used to arrange the objects (that is, the cases), while another was used to encode the objects' values. Examples of this notation appear in Table 23.8 and Figure 23.21.

TABLE 23.8
Taxonomy of Information Visualization Techniques



I. Simple Visual Structures

Direct Reading
1-Variable [X]
- Lists
- 1D object charts
- 1D scatterplots
- Pie charts
- Folded dimensions
- Distributions
- Box plots

2-Variable [XY]
- 2D object charts
- 2D scatterplots

3-Variable
- [XYR]
  - Retinal scatterplot
  - Kahonen diagrams
  - Retinal topographies
- [(XY)Z]
  - Information landscapes
  - Information surfaces
- [XYZ]
  - 3D scatterplots

4-Variable
- [XYZR]
  - 3D retinal scatterplots
  - 3D topographies

—Barrier of Perception—

Articulated Reading
n-Variable
- [XYR^{n-2}]
  - 2D Retinal scatterplots
- [XYZR^{n-3}]
  - 2D Retinal scatterplots

Trees
- Node and link trees
- Enclosure trees
- TreeMaps
- Cone trees

Networks
Time

II. Composed Visual Structures

Single-axis Composition [XY^n]
- Permutation matrices
- Parallel coordinates

Double-axis Composition [XY]
- Graphs

Recursive Composition
- 2D in 2D [(XY)^{XY}]
  - Scatterplot matrices
  - Prosection matrices
  - Hierarchical axes
- Marks in 2D [(XY)^R]
  - Stick figures
  - Color icons
  - Shape coding
  - Keim spirals
- 3D in 3D [(XYZ)^{XYZ}]
  - Worlds within worlds

III. Interactive Visual Structures

- Dynamic queries
- Magic lens
- Overview + detail
- Linking and brushing
- Extraction & comparison
- Attribute explorer

IV. Focus + Context Attention-reactive Visual Abstraction

Data-based Methods
- Filtering
- Selective aggregation

View-based Methods
- Micro-macro readings
- Highlighting
- Visual transfer functions
- Perspective distortion
- Alternate geometries

23.3 SIMPLE VISUAL STRUCTURES

The design of information visualizations begins with mappings from variables of the Data Table into the Visual Structure. The basic strategy for the visualization designer could be described as follows:

1. *Determine which variables of the Analytic Abstraction to map into spatial position in the Visual Structure.*
2. *Combine these mappings to increase dimensionality (e.g., by folding).*
3. *Use retinal variables as an overlay to add more dimensions.*
4. *Add controls for interaction.*
5. *Consider attention-reactive features to expand space and manage attention.*

We start by considering some of the ways in which variables can be mapped into space.

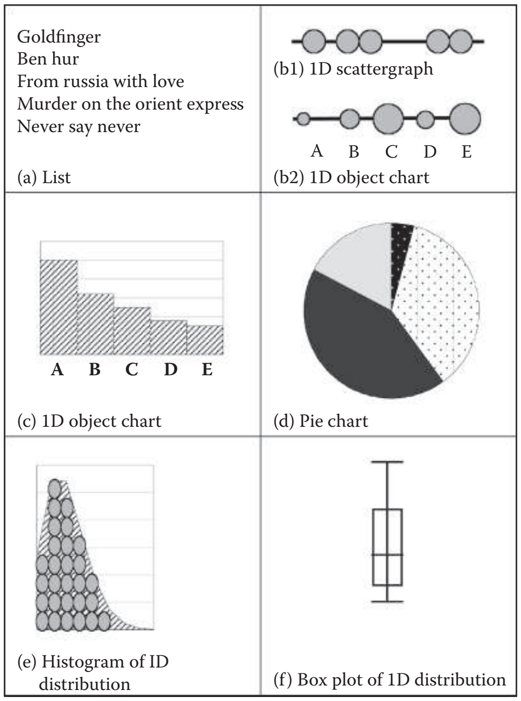
23.3.1 1-Variable

One-variable visual displays may actually use more than one visual dimension. This is because the data variable or attribute is displayed against some set of objects using some mark and because the mark itself takes space. Or, more subtly, it may be because one of the dimensions is used for arranging the objects and another for encoding via position of the variable. A simple example would be when the data are just visually mapped into a simple text list as in Figure 23.14a. The objects form a sequence on the Y-dimension, and the width of the marks (the text descriptor) takes space in the X-dimension. By contrast, a one-dimensional scattergraph (Figure 23.14b) does not use a dimension for the objects. Here, the Y-axis is used to display the attribute variable (suppose these are distances from home of gas stations); the objects are encoded in the mark (which takes a little bit of the X-dimension).

More generally, many single-variable visualizations are in the form $v = f(o)$, where v is a variable attribute and o is the object. Figure 23.14c is of this form and uses the Y-axis to encode the variable and the X-axis for the objects. Note that if the objects are, as usual, nominal, then they are reorderable: sorting the objects on the variable produces easily perceivable visual patterns. For convenience, we have used rectangular coordinates, but any other orthogonal coordinates could be used as the basis of decomposing space. Figure 23.14d uses θ from polar coordinates to encode, say, percentage voting for different presidential candidates. In Figure 23.14e, a transformation on the data side has transformed variable o into a variable representing the distribution, then mapped that onto points on the Y-axis. In Figure 23.14f, another transformation on the data side has mapped this distribution into 2nd quartiles, 3rd quartiles, and outlier points, which is then mapped on the visual side into a box plot on the Y-axis. Simple as they are, these techniques can be very useful, especially in combination with other techniques.

One special, but common, problem is how to visualize very large dimensions. This problem occurs for single-variable visualizations, but may also occur for one dimension of a multi-variable visualization. Figure 23.15 shows several techniques for handing the problem. In Figure 23.15a (Freeman and Fertig 1995), the visual dimension is laid out in perspective. Even though each object may take only one or a few pixels on the axis, the objects are actually fairly large and selectable in the diagram. In Figure 23.15b (Eick, Steffen, and Sumner 1992), the objects (representing lines of code) are laid out on a *folded* Y-axis. When the Y-axis reaches the bottom of the page, it continues offset at the top. In Figure 23.15c (Keim and Kriegel 1994), the axis is wrapped in a square spiral. Each object is a single pixel, and its value is coded as the retinal variable color hue. The objects have been sorted on another variable; hence, the rings show the correlation of this attribute with that of the sorting attribute.

One-variable visualizations are also good parts of controls. Controls, in the form of slides, also consume considerable space on the display (for example, the controls in Figure 23.1) that could be used for additional information communication. Figure 23.15d shows a slider on whose surface is a distribution representation of the number of objects for each value of the input variable, thereby communicating information about the slider's sensitivity in different data ranges. The slider on the left of Figure 23.15b has a one-variable visualization that serves as a legend for the main visualization: it associates color hues with dates and allows the selection of date ranges.



(a) List

(b1) 1D scattergraph

(b2) 1D object chart

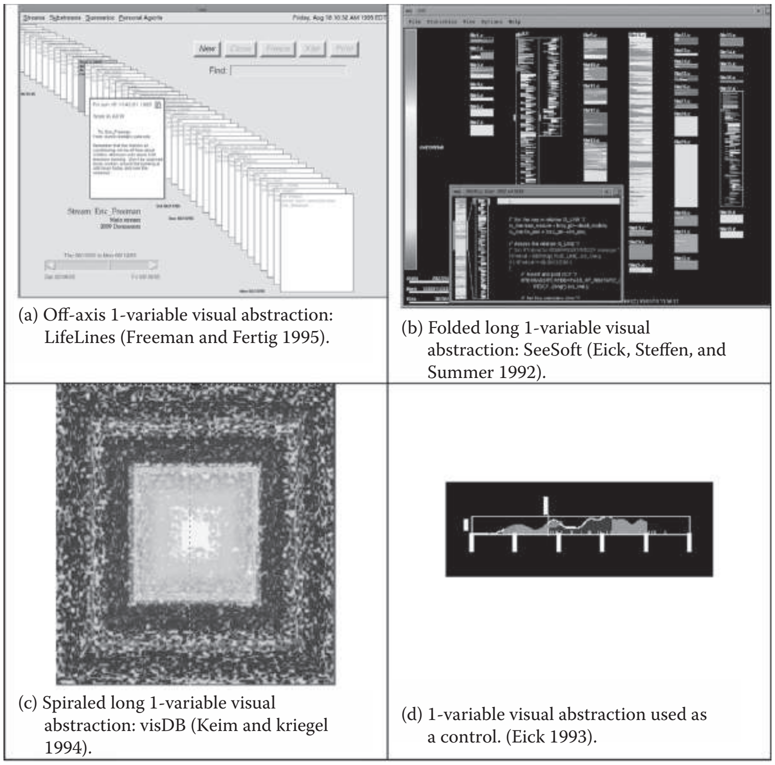
(c) 1D object chart

(d) Pie chart

(e) Histogram of ID distribution

(f) Box plot of 1D distribution

FIGURE 23.14 1-variable visual abstractions.



(a) Off-axis 1-variable visual abstraction: LifeLines (Freeman and Fertig 1995).

(b) Folded long 1-variable visual abstraction: SeeSoft (Eick, Steffen, and Summer 1992).

(c) Spiraled long 1-variable visual abstraction: visDB (Keim and kriegel 1994).

(d) 1-variable visual abstraction used as a control. (Eick 1993).

FIGURE 23.15 Uses of 1-variable visual abstractions.

23.3.2 2-Variables

As we increase the number of variables, it is apparent that their mappings form a combinatorial design space. Figure 23.16 schematically plots the structure of this space, leaving out the use of multiple lower variable diagrams to plot higher variable combinations. Two-variable visualizations can be thought of as a composition of two elementary axes (Bertin 1977, 1981; Mackinlay 1986b), which use a single mark to encode the position on both those axes. Mackinlay called this *mark composition*, and it results in a 2D scattergraph (Figure 23.16g). Note that instead of mapping onto two positional visual encodings, one positional axis could be used for the objects, and the data variables could be mapped onto a position encoding and a retinal encoding (size), as in Figure 23.16f.

23.3.3 3-Variables and Information Landscapes

By the time we get to three data variables, a visualization can be produced in several ways. We can use three separate visual dimensions to encode the three data variables in a *3D scattergraph* (Figure 23.16j). We could also use two spatial dimensions and one retinal variable in a *2D retinal scattergraph* (Figure 23.16k). Or we could use one spatial dimension as an object dimension, one as a data attribute dimension, and two retinal encodings for the other variables, as in an *object chart* such as in Figure 23.16i. Because Figure 23.16i uses multiple retinal encodings, however, it may not be as effective as other techniques. Notice that because they all encode three data variables, we have classified 2D and 3D displays together. In fact, one popular 3-variable information visualization that lies between 2D and 3D is the *information landscape* (Figure 23.16m). This is essentially a 2D scattergraph with one data variable extruded into the third spatial dimension. Its essence is that two of the spatial dimensions are more tightly coupled and often relate to a 2D visualization. For example, the two dimensions might form a map with the bars showing the GDP of each region.

Another special type of 3-variable information visualization is a 2D *information topography.* In an information typography, space is partly defined by reference to external structure. For example, the topography of Figure 23.17a is a map of San Francisco, requiring two spatial variables. The size of blue dots indexes the number of domain names registered to San Francisco street addresses. Looking at the patterns in the visualization shows that Internet addresses have especially concentrated in the Mission and South of Mission districts. Figure 23.17a uses a topography derived from real geographical space. Various techniques, such as multidimensional scaling, factor analysis, or connectionist self-organizing algorithms, can create abstract spaces based on the similarities among collections of documents or other objects. These abstract similarity spaces can function like a topography. An example can be seen in Figure 23.17b, where the pages in a website are depicted as regions in a similarity space. To create this diagram,* a web crawler crawls the site and indexes all the words and pages on the site. Each page is then turned into a document vector to represent the semantic content of that page. The regions are created using a neural network learning algorithm (see Lin, Soergel, and Marchionini 1991). This algorithm organizes the set of web pages into regions. A visualization algorithm then draws boundaries around the regions, colors them, and names them. The result, called a *Kahonen diagram* after its original inventor, is a type of *retinal similarity topography.*

Information landscapes can also use marks that are surfaces. In Figure 23.18a, topics are clustered on a similarity surface, and the strength of each topic is indicated by a 3D contour. A more extreme case is Figure 23.18b, where an information landscape is established in spherical coordinates, and the amount of ozone is plotted as a semitransparent overlay on the ρ-axis.

23.3.4 *n*-Variables

Beyond three variables, direct extensions of the methods we have discussed become less effective. It is possible, of course to make plots using two spatial variables and *n*–2 retinal variables, and the possibilities for four variables are shown in Figure 23.16. These diagrams can be understood, but at the cost of progressively more effort as the number of variables increases. It would be very difficult to understand an [XYR^{20}] retinal scattergraph, for example.

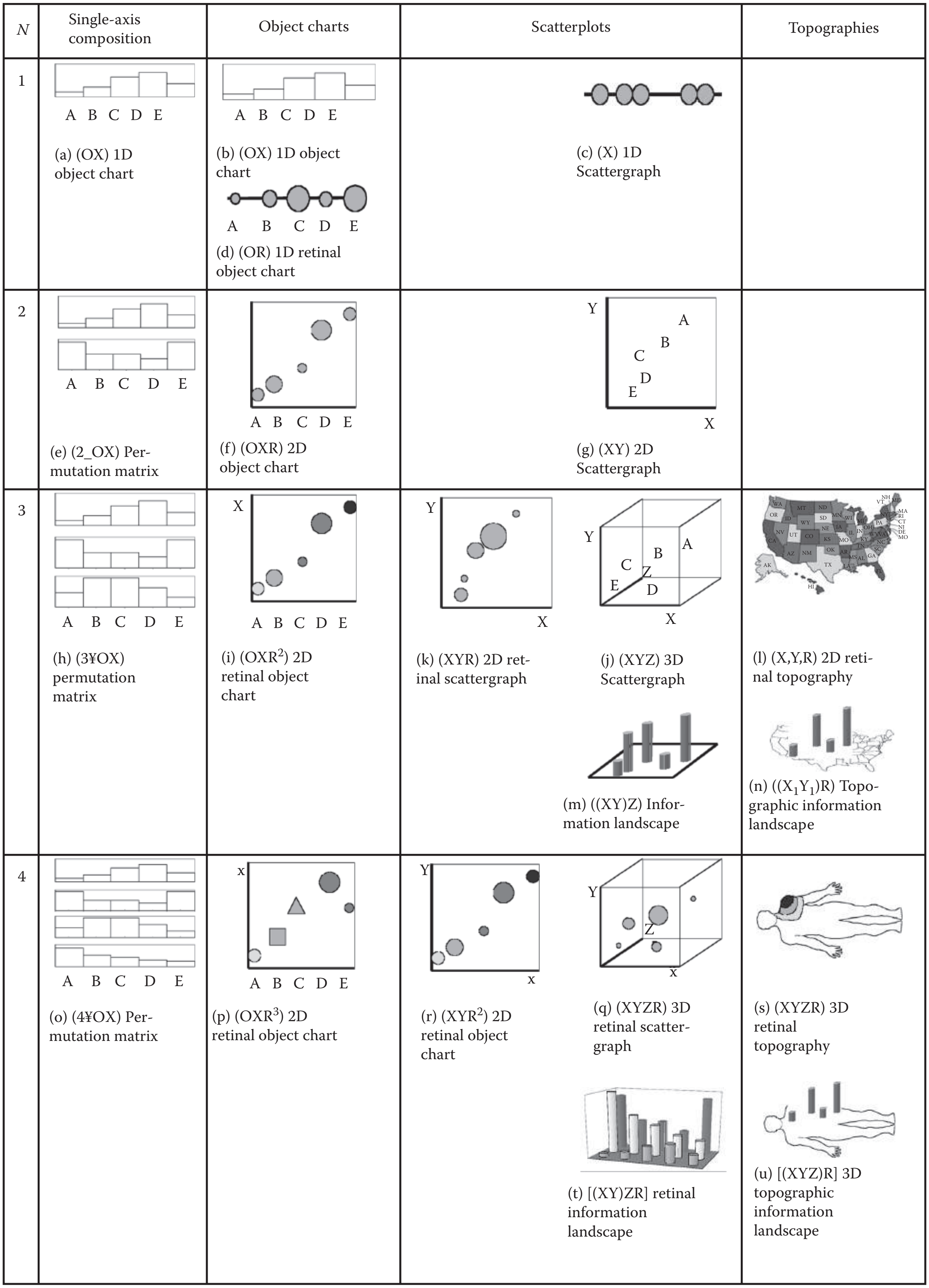
23.3.5 Trees

An interesting alternative to showing variable values by spatial positioning is to use explicitly drawn linkages of some kind. Trees are the simplest form of these. Trees map cases into subcases. One of the data variables in a Data Table (for example, the variable Reports To in an organization chart) is used to define the tree. There are two basic methods for visualizing a tree: (1) Connection and (2) Enclosures.

23.3.5.1 Connection

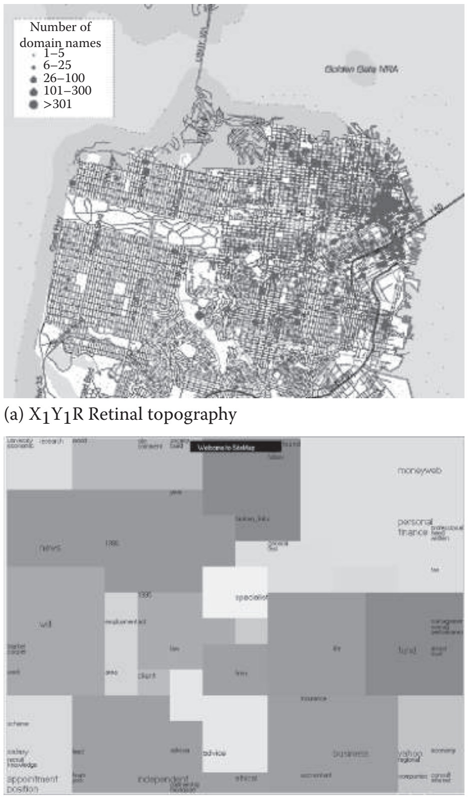
Connection uses lines to connect marks signifying the nodes of the tree. Logically, a tree could be drawn merely by drawing lines between objects randomly positioned on the plane, but such a tree would be visually unreadable. Positioning in space is important. Figure 23.19a is a tree from Charles Darwin's notebook (Robin 1992) drawn to help him work out the theory of evolution. Lines proceed from ancestor species to new species. Note that even in this informal setting intended for personal use that the tree uses space systematically (and opportunistically). There are no crossed lines. A common way of laying out trees is to have the depth in the tree map onto one ordinal access, as in Figure 23.19b, while the other axis is nominal and used to separate nodes.

* This figure is produced by a program called SiteMap by Xia Lin and Associates. See http://faculty.cis.drexel.edu/sitemap/index.html.



N	Single-axis composition	Object charts	Scatterplots	Topographies
1	(a) (OX) 1D object chart	(b) (OX) 1D object chart (d) (OR) 1D retinal object chart	(c) (X) 1D Scattergraph	
2	(e) (2_OX) Permutation matrix	(f) (OXR) 2D object chart	(g) (XY) 2D Scattergraph	
3	(h) (3¥OX) permutation matrix	(i) (OXR^2) 2D retinal object chart	(k) (XYR) 2D retinal scattergraph (j) (XYZ) 3D Scattergraph (m) ((XY)Z) Information landscape	(l) (X,Y,R) 2D retinal topography (n) ((X$_1$Y$_1$)R) Topographic information landscape
4	(o) (4¥OX) Permutation matrix	(p) (OXR^3) 2D retinal object chart	(r) (XYR^2) 2D retinal object chart (q) (XYZR) 3D retinal scattergraph (t) [(XY)ZR] retinal information landscape	(s) (XYZR) 3D retinal topography (u) [(XYZ)R] 3D topographic information landscape

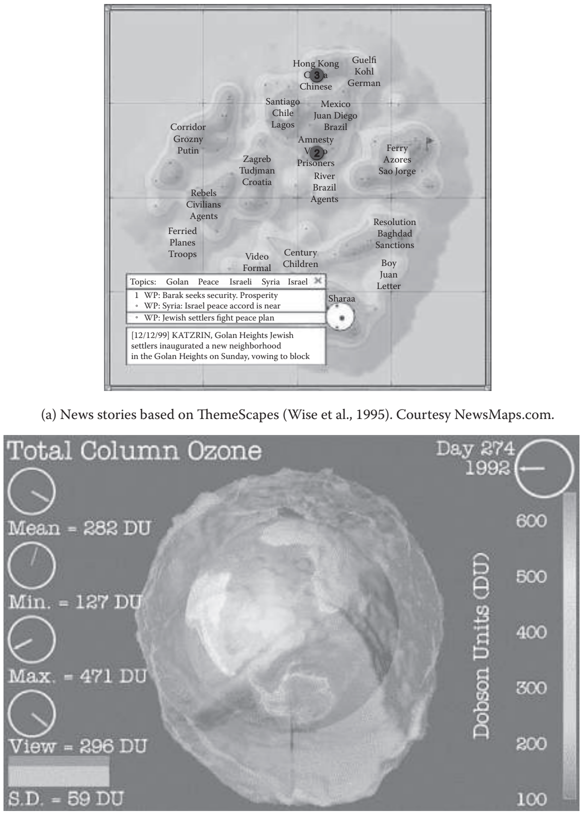
FIGURE 23.16 Simple visual structures.



(a) X_1Y_1R Retinal topography

(b) X_SY_SR Retinal similarity topography

FIGURE 23.17 Retinal information topographies.



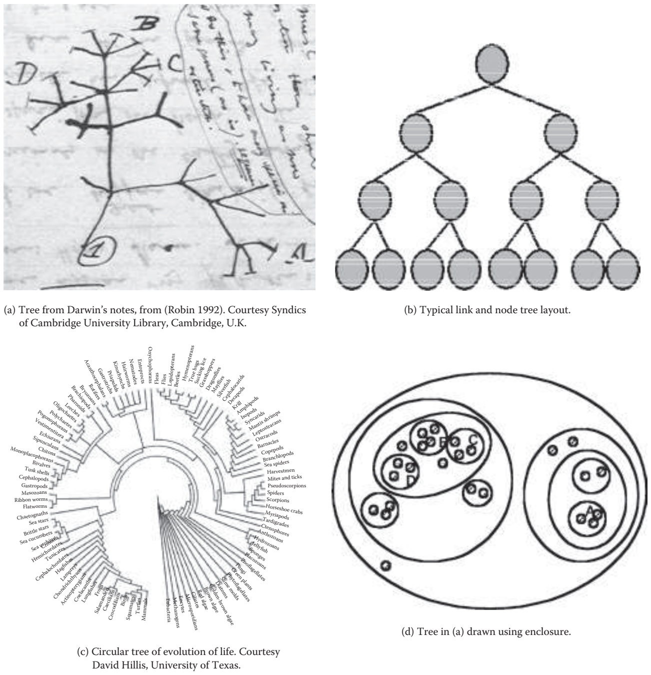
1 WP: Barak seeks security. Prosperity
WP: Syria: Israel peace accord is near
WP: Jewish settlers fight peace plan

[12/12/99] KATZRIN, Golan Heights Jewish settlers inaugurated a new neighborhood in the Golan Heights on Sunday, vowing to block

(a) News stories based on ThemeScapes (Wise et al., 1995). Courtesy NewsMaps.com.

(b) Ozone layer surrounding earth. L. Treinish. Courtesy IBM.

FIGURE 23.18 3D information surface topographies.



(a) Tree from Darwin's notes, from (Robin 1992). Courtesy Syndics of Cambridge University Library, Cambridge, U.K.

(b) Typical link and node tree layout.

(c) Circular tree of evolution of life. Courtesy David Hillis, University of Texas.

(d) Tree in (a) drawn using enclosure.

FIGURE 23.19 Trees.

Of course, trees could also be mapped into other coordinate systems: for example, there can be circular trees in which the *r*-axis represents depth and the θ-axis is used to separate nodes as in the representation of the evolution species in Figure 23.19c.* It is because trees have no cycles that one of the spatial dimensions can be used to encode tree depth. This partial correlation of tree structure and space makes trees relatively easy to lay out and interpret, compared to generalized networks. Hierarchical displays are important not only because many interesting collections of information, such as organization charts or taxonomies, are hierarchical data, but also because important collections of information, such as websites, are *approximately* hierarchical. Whereas practical methods exist for displaying trees up to several thousand nodes, no good methods exist for displaying general graphs of this size. If a visualization problem involves the displaying of network data, a practical design heuristic is to see whether the data might not be forced into a display as a modified tree, such as a tree with a few non-tree links. A significant disadvantage of trees is that as they get large, they acquire an extreme aspect ratio, because the nodes expand exponentially as a function with depth. Consequently, any sufficiently large tree (say, >1000 nodes) resembles a straight line. Circular trees such as Figure 23.19c are one way of trying to buy more space to mitigate this problem. Another disadvantage of trees is the significant empty space between nodes to make their organization easily readable. Various tricks can be used to wrap parts of the tree into this empty space, but at the expense of the tree's virtues of readability.

* This figure is from David Hillis, University of Texas.

23.3.5.2 Enclosure

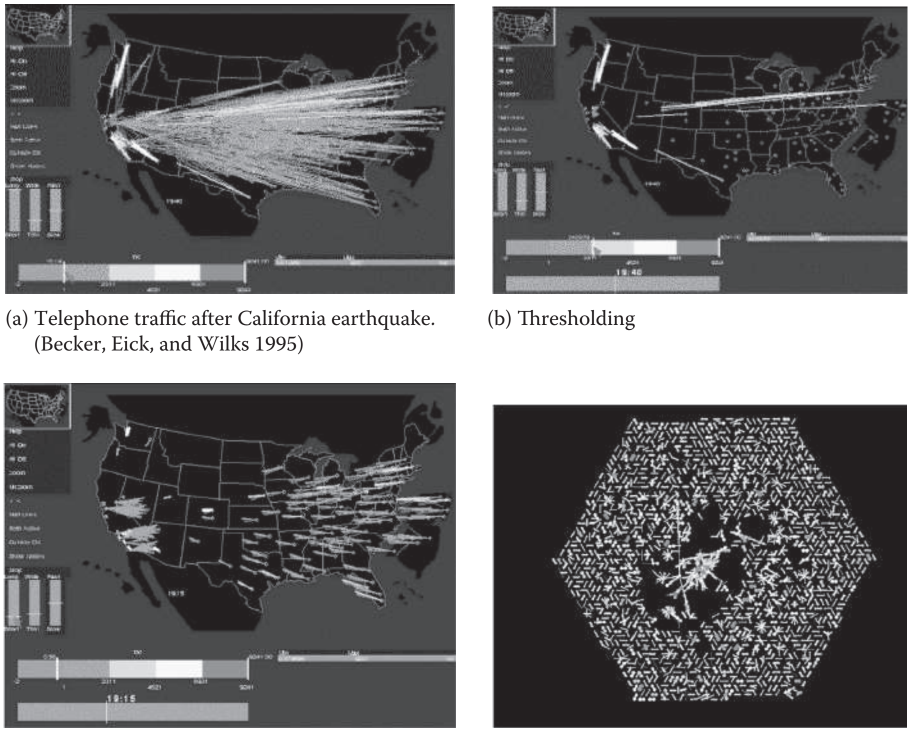
Enclosure uses lines to hierarchically enclose nested subsets of the tree. Figure 23.19d is an enclosure tree encoding of Darwin's tree in Figure 23.19a. We have already seen one attempt to use tree enclosure, TreeMaps (Figure 23.5). TreeMaps make use of all the space and stay within prescribed space boundaries, but they do not represent the nonterminal nodes of the tree very well and similar leaves can have wildly different aspect ratios. Recent variations on TreeMaps found ways to "squarify" nodes (Shneiderman and Wattenberg 2001), mitigating this problem.

23.3.6 Networks

Networks are more general than trees and may contain cycles. Networks may have directional links. They are useful for describing communication relationships among people, traffic in a telephone network, and the organization of the Internet. Containment is difficult to use as a visual encoding for network relationships, so most networks are laid out as node and link diagrams. Unfortunately, straightforward layouts of large node and link diagrams tend to resemble a large wad of tangled string.

We can distinguish the same types of nodes and links in network Visual Structures that we did for spatial axes: (a) Unstructured (unlabeled), (b) Nominal (labeled), (c) Ordinal (labeled with an ordinal quantity), or (d) Quantitative (weighted links). Retinal properties, such as size or color, can be used to encode information about links and nodes. As in the case of trees, spatial positioning of the nodes is extremely important. Network visualizations escape from the strong spatial constraints of simple Visual Structures only to encounter another set of strong spatial constraints of node links crossing and routing. Networks and trees are not so much an alternative of the direct graphical mappings we have discussed so far as they are another set of techniques that can be overlaid on these mappings. Small node and link diagrams can be laid out opportunistically by hand or by using graph drawing algorithms that have been developed (Battista et al. 1994; Cruz and Tamassia 1998; Tamassia 1996) to optimize minimal link crossing, symmetry, and other aesthetic principles.

For very large node and link diagrams, additional organizing principles are needed. If there is an external topographic structure, it is sometimes possible to use the spatial variables associated with the nodes. Figure 23.20a shows a network based on call traffic between cities in the United States (Becker, Eick, and Wilks 1995). The geographical location of the cities is used to lay out the nodes of the network. Another way to position nodes is by associating nodes with positions in a similarity space, such that the nodes that have the strongest linkages to each other are closest together. There are several methods for computing node nearness in this way. One is to use multidimensional scaling (MDS) (Fairchild, Poltrock, and Furnas 1988). Another is to use a "spring" technique, in which each link is associated with a Hooke's Law spring weighted by strength of association and the system of springs is solved to obtain node



(a) Telephone traffic after California earthquake. (Becker, Eick, and Wilks 1995)

(b) Thresholding

(c) Line shortening

(d) Visualization to detect telephone fraud. (Cox, Eick, and Wills 1997)

FIGURE 23.20 Network methods.

position. Eick and Willis (1993) have argued that the MDS technique places too much emphasis on smaller links. They have derived an alternative that gives clumpier (and hence, more visually structured) clusters of nodes. If positioning of nodes corresponds perfectly with linkage information, then the links do not add more visual information. If positioning does not correspond at all with linkage information, then the diagram is random and obscure. In large graphs, node positions must have a partially correlated relationship to linkage in order to allow the emergence of visual structure. Note that this is what happens in the telephone traffic diagram, Figure 23.20a. Cities are positioned by geographical location. Communication might be expected to be higher among closer cities, so the fact that communications is heavy between coasts stands out.

A major problem in a network such as Figure 23.20a is that links may obscure the structure of the graph. One solution is to route the links so that they do not obscure each other. The links could even be drawn outside the plane in the third dimension; however, there are limits to the effectiveness of this technique. Another solution is to use *thresholding*, as in Figure 23.20b. Only those links representing traffic greater than a certain threshold are included; the others are elided, allowing us to see the most important structure. Another technique is *line shortening*, as in Figure 23.20c. Only the portion of the line near the nodes is drawn. At the cost of giving up the precise linkage, it is possible to read the density of linkages for the different nodes. Figure 23.20d is a technique used to find patterns in an extremely large network. Telephone subscribers are represented as nodes on a hexagonal array. Frequent pairs are located near each other on the array. Suspicious patterns are visible because of the sparseness of the network.

The insightful display of large networks is difficult enough that many information visualization techniques depend on interactivity. One important technique, for example, is node aggregation. Nodes can be aggregated to reduce the number of links that have to be drawn on the screen. Which nodes are aggregated can depend on the portion of the network on which the user is drilling down. Similarly, the sets of nodes can be interactively restricted (e.g., telephone calls greater than a certain volume) to reduce the visualization problem to one within the capability of current techniques.

23.4 COMPOSED VISUAL STRUCTURES

So far, we have discussed simple mappings from data into spatial position axes, connections and enclosures, and retinal variables. These methods begin to run into a barrier around three variables as the spatial dimensions are used up and as multiples of the less efficient retinal variables needed. Most interesting problems involve many variables. We shall therefore look at a class of methods that reuse precious spatial axes to encode variables. This is done by composing a compound Visual Structure out of several simple Visual Structures. We will consider five subclasses of such composition: (a) single-axis composition, (b) double-axis composition, (c) mark composition, (d) case composition, and (e) recursive composition. Schematically, we illustrate these possibilities in Figure 23.21.

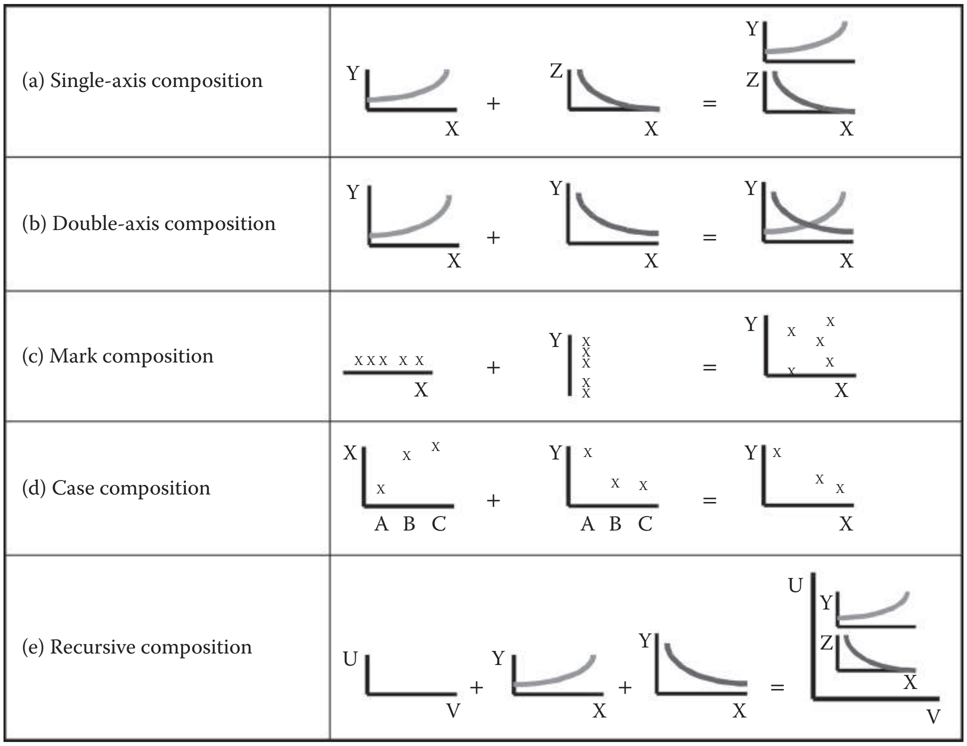


FIGURE 23.21 Composition types.

23.4.1 Single-Axis Composition

In single-axis composition, multiple variables that share a single axis are aligned using that axis, as illustrated in Figure 23.21a. An example of single-axis composition is a method due to Bertin called *permutation matrices* (Bertin 1977/1981). In a permutation matrix (Figure 23.16o, for example), one of the spatial axes is used to represent the cases and the other a series of bar charts (or rows of circles of different size or some other depiction of the value of each variable) to represent the values. In addition, bars for values below average may be given a different color, as in Figure 23.7, in order to enhance the visual patterns. The order of the objects and the order of the variables may both be permuted until patterns come into play. Permutation matrices were used in our hotel analysis example. They give up direct reading of the data space in order to handle a larger number of variables. Of course, as the number of variables (or objects) increases, manipulation of the matrices becomes more time-consuming and visual interpretation more complex. Still, permutation matrices or their variants are one of the most practical ways of representing multi-variable data.

If we superimpose the bar charts of the permutation matrix atop one another, and then replace the bar chart with a line linking together the tops of the bars, we get another method for handling multiple variables by single-axis composition—*parallel coordinates* (Inselberg 1997; Inselberg and Dimsdale 1990), as shown in Figure 23.22. A problem is analyzed in parallel coordinates by interactively restricting the objects displayed (the lines) in order to look at cases with common characteristics. In Figure 23.22, parallel coordinates are used to analyze the problem of yield from a certain processor chip. X1 is chip yield, X2 is quality, X3 through X12 are defects, and the rest of the variables are physical parameters. The analysis, looking at those subsets of data with high yield and noticing the distribution of lines on the other parameters, was able to solve a significant problem in chip processing.

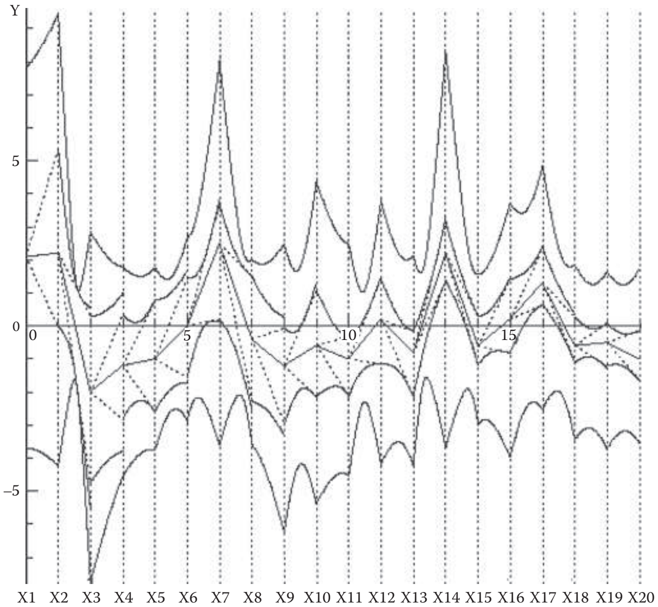


FIGURE 23.22 Single-axis composition: parallel coordinates.

Both permutation matrices and parallel coordinates allow analyses in multi-dimensional space, because they are efficient in the use (and reuse) of spatial position and the plane. Actually, they also derive part of their power from being interactive. In the case of permutation matrices, interactivity comes in reordering the matrices. In the case of parallel coordinates, interactivity comes in selecting subsets of cases to display.

23.4.2 Double-Axis Composition

In double-axis composition, two visual axes must be in correspondence, in which case the cases are plotted on the same axes as a multivariable graph (Figure 23.21b). Care must be taken that the variables are plotted on a comparable scale. For this reason, the separate scales of the variables are often transformed to a common proportion change scale. An example would be change in price for various stocks. The cases would be the years, and the variables would be the different stocks.

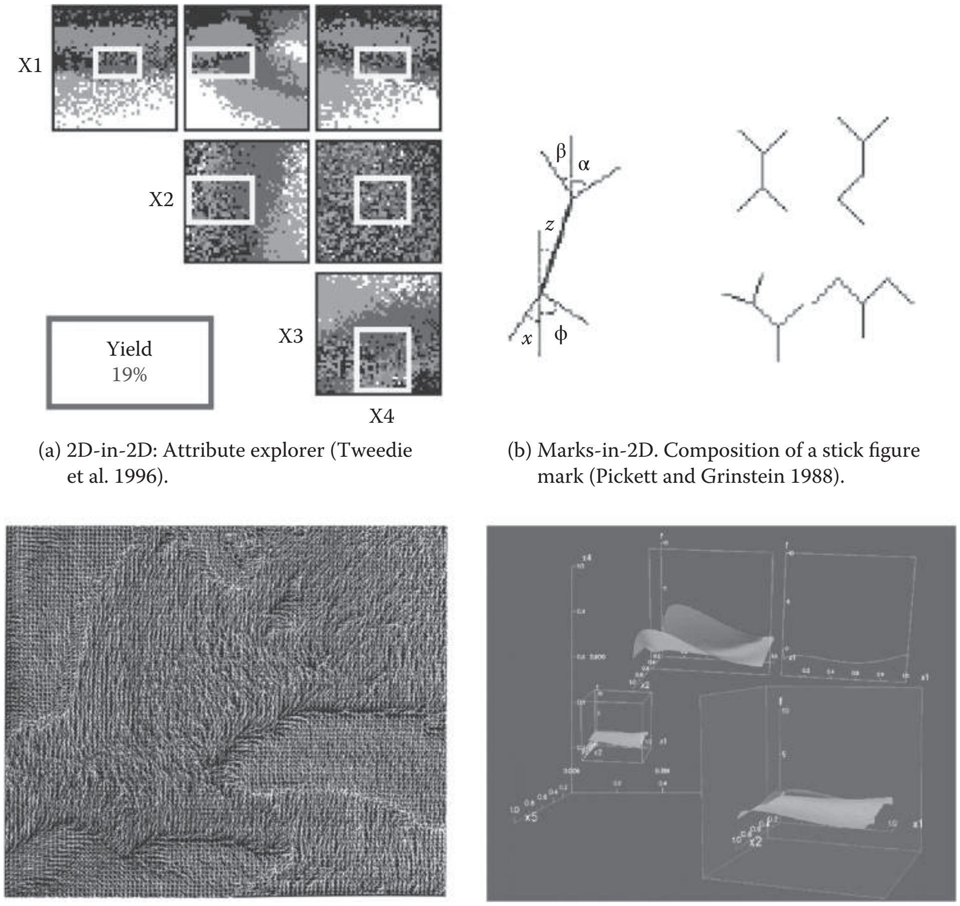
23.4.3 Mark Composition and Case Composition

Composition can also fuse diagrams. We discussed that each dimension of visual space can be said to have properties, as summarized in Table 23.4. The visual space of a diagram is composed from the properties of its axis. In *mark composition* (Figure 23.21c), the mark on one axis can fuse with the corresponding mark on another axis to form a single mark in the space formed by the two axes. Similarly, two object charts can be fused into a single diagram by having a single mark for each case. We call this latter form *case composition* Figure 23.21d.

23.4.4 Recursive Composition

Recursive composition divides the plane (or 3D space) into regions, placing a subvisualization in each region (Figure 23.21e). We use the term somewhat loosely, since regions have different types of subvisualizations. The FilmFinder in Figure 23.1 is a good example of a recursive visualization. The screen breaks down into a series of simple Visual Structures and controls: (a) a 3-variable retinal scattergraph (Year, Rating, FilmType) + (b) a 1-variable slider (Title) + (c) a 1-variable slider (Actors) + (d) a 1-variable slider (Actresses) + (e) a 1-variable slider (Director) + (f) a 1-variable slider (FilmLength) + (g) a 1-variable radio button control (Rating) + (h) a 1-variable button-set (FilmType).

Three types of recursive composition deserve special mention: (a) 2D-in-2D, (b) marks-in-2D, and (c) 3D-in-3D. An example of *2D-in-2D composition* is the "prosection matrix" (Tweedie et al. 1996) shown in Figure 23.23a. Each smaller square in the prosection matrix represents a pair of parameters plotted against each other. The coloring shows which values of the plotted pair give excellent (red region) or partly good (gray regions) performance for the design of some device. The arrangement of the individual matrices into a supermatrix redefines the spatial dimensions (that is, associates it



(a) 2D-in-2D: Attribute explorer (Tweedie et al. 1996).

(b) Marks-in-2D. Composition of a stick figure mark (Pickett and Grinstein 1988).

(c) Visualization of stick figures showing weather around Lake Ontario.

(d) 3D-in-3D: Worlds-within-worlds (Feiner and Beshers 1990).

FIGURE 23.23 Recursive composition.

with different variables) within each of the cells, and the cells themselves are arranged in an overall scheme that systematically uses space. In this way, the precious spatial dimension is effectively expanded to where all the variables can reuse it. An important property of techniques similar to this one is that space is defined at more than one *grain size*, and these levels of grain become the basis for a *macro-micro reading.*

An example of *marks-in-2D composition* is the use of “stick figure” displays. This is an unusual type of visualization in which the recursion is within the mark instead of within the use of space. Figure 23.23b shows a mark that is itself composed of submarks. The mark is a line segment with four smaller line segments protruding from the ends. Four variables are mapped onto angles of these smaller line segments and a fifth onto the angle of the main line segment. Two additional variables are mapped onto the position of this mark in a 2D display. A typical result is the visualization in Figure 23.23c, which shows five weather variables around Lake Ontario, the outline of which clearly appears in the figure.

Feiner and Beshers (1990) provided an example of the third recursive composition technique, *3D-in-3D composition*. Suppose a dependent variable is a function of six continuous variables, $y = f(x, y, z, w, r, s)$. Three of these variables are mapped onto a 3D coordinate system. A position is chosen in that space, say, $x1$, $y1$, $z1$. At that position, a new 3D coordinate system is presented with a surface defined by the other three variables (Figure 23.23d). The user can thus view $y = f(x1, y1, z1, w, r, s)$. The user can slide the second-order coordinate system to any location in the first, causing the surface to change appropriately. Note that this technique combines a composed visual inter-action with interactivity on the composition. Multiple second-order coordinate systems can be displayed at the space simultaneously, as long as they do not overlap by much.

23.5 INTERACTIVE VISUAL STRUCTURES

In the examples we have considered so far, we have often seen that information visualization techniques were enhanced by being interactive. Interactivity is what makes visualization a new medium, separating it from generations of excellent work on scientific diagrams and data graphics. Interactivity means controlling the parameters in the visualization reference model (Figure 23.10). This naturally means that there are different types of interactivity, because the user could control the parameters to data transformations, to visual mappings, or to view transformations. It also means that there are different forms of interactivity based on the response cycle of the interaction. As an approximation, we can think of there being three time constants that govern interactivity, which we take to be 0.1 sec, 1 sec, and 10 sec (Card, Moran, and Newell 1986) (although the ideal value of these may be somewhat less, say, 0.07 sec, 0.7 sec, and 7 sec). The first time constant is the time in which a system response must be made, if the user is to feel that there is a direct physical manipulation of the visualization. If the user clicks on a button or moves a slider, the system needs to update the display in less than 0.1 sec. Animation frames need to take less than 0.1 sec. The second time constant, 1 sec, is the time to complete an immediate action, for example, an animated sequence such as zooming in to the data or rotating a tree branch. The third time constant 10 sec (meaning somewhere in the 5 to 30 sec interval) is the time for completing some cognitive action, for example deleting an element from the display. Let us consider a few well-known techniques for interactive information visualizations.

23.5.1 Dynamic Queries

A general paradigm for visualization interaction is dynamic queries, the interaction technique used by the FilmFinder in Figure 23.1. The user has a visualization of the data and a set of controls, such as sliders, by which subsets of the Data Table can be selected. For example, Table 23.9 shows the mappings of the Data Table and controls for the FilmFinder. The sliders and other controls will select which subset of the data is going to be displayed. In the FilmFinder, the control for Length is a two-sided slider. Setting one end to 90 minutes and the other end to 120 minutes will select for display only those cases of the Data Table whose year variable lies between these limits. The display needs to change within the 0.1 sec of changing the slider.

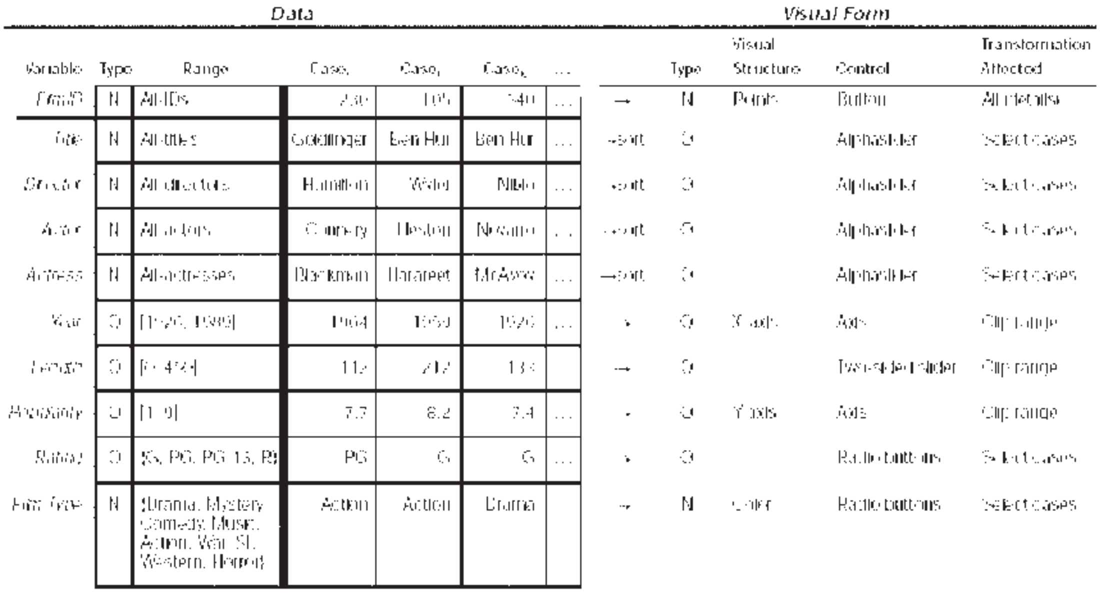
23.5.2 Magic Lens (Movable Filter)

Dynamic queries is one type of interactive filter. Another type is a movable filter that can be moved across the display, as in Figure 23.24a. These *magic lenses* are useful when it is desired to filter only some of the display. For example, a magic lens could be used with a map that showed the population of any city it was moved over. Multiple magic lenses can be used to cascade filters.

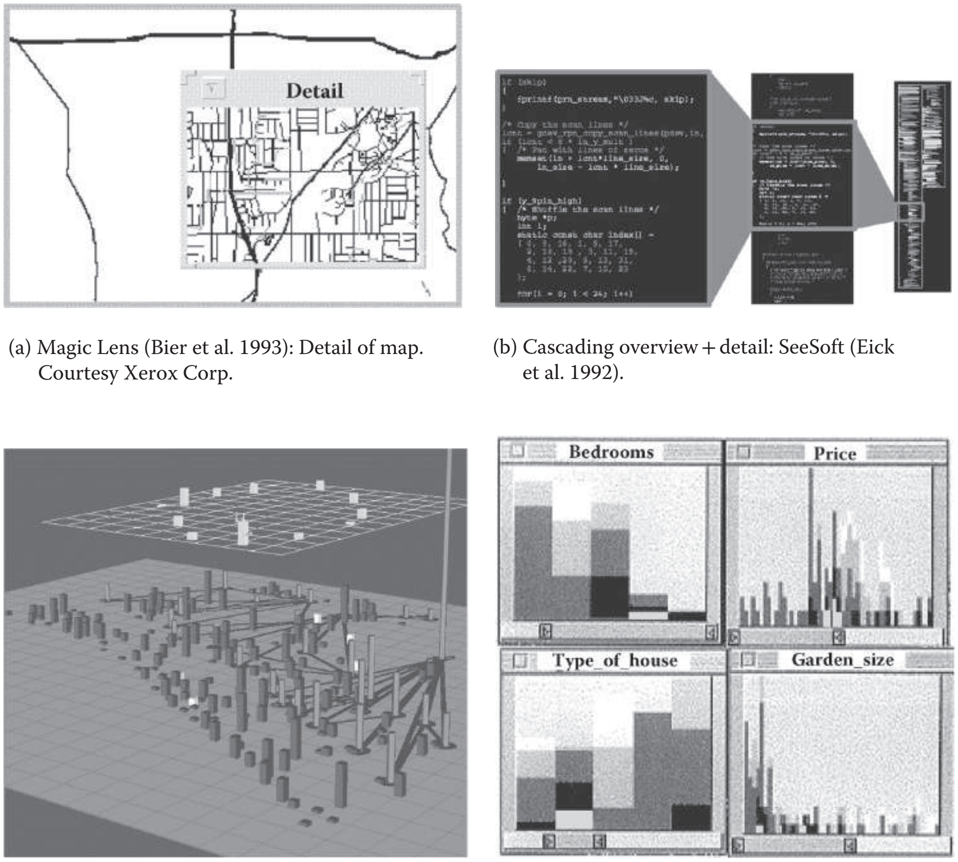
23.5.3 Overview Detail

We can think of an overview + detail display (Figure 23.24b) as a particular type of magic lens, one that magnifies the display and has the magnified region off to the side so as not to occlude the region. Displays have information at different grain sizes. A GIS map may have information at the level of a continent as well as at the level of a city. If the shape of the continent can be seen, the display is too coarse to see the roadways of a city. Overview + detail displays show that data at more than one level, but they also show where the finer grain display fits into the larger grain display. In Figure 23.24b, from SeeSoft (Eick, Steffen, and Sumner 1992), a system for visualizing large software systems, the amount of magnification in the detail view is large enough that two concatenated overview + detail displays are required. Overview

TABLE 23.9
Visual Marks and Controls for FilmFinder



Data							Visual Form				
Variable	Type	Range	$Case_i$	$Case_j$	$Case_k$	...		Type	Visual Structure	Control	Transformation Affected
FilmID	N	All IDs	230	105	340	...	→	N	Points	Button	All (details)
Title	N	All titles	Goldfinger	Ben Hur	Ben Hur	...	→sort	O		Alphaslider	Select cases
Director	N	All directors	Hamilton	Wyler	Niblo	...	→sort	O		Alphaslider	Select cases
Actor	N	All actors	Connery	Heston	Novarro	...	→sort	O		Alphaslider	Select cases
Actress	N	All actresses	Blackman	Harareet	McAvoy	...	→sort	O		Alphaslider	Select cases
Year	Q	[1926, 1989]	1964	1959	1926	...	→	Q	X axis	Axis	Clip range
Length	Q	[0, 434]	112	212	133		→	Q		Two-sided slider	Clip range
Popularity	Q	[1, 9]	7.7	8.2	7.4	...	→	Q	Y axis	Axis	Clip range
Rating	O	{G, PG, PG-13, R}	PG	G	G	...	→	O		Radio buttons	Select cases
Film Type	N	{Drama, Mystery, Comedy, Music, Action, War, SF, Western, Horror}	Action	Action	Drama		→	N	Color	Radio buttons	Select cases



```
if (skip)
{
   fprintf(prn_stream,"\033J%c", skip);
}

/* Copy the scan lines */
lcnt = gdev_prn_copy_scan_lines(pdev,in,
if (lcnt < 8 * in_y_mult )
{  /* Pad with lines of zeros */
   memset(in + lcnt*line_size, 0,
       in_size - lcnt * line_size);
}

if (y_9pin_high)
{  /* Shuffle the scan lines */
   byte *p;
   int i;
   static const char index[] =
   { 0, 8, 16, 1, 9, 17,
     2, 10, 18 , 3, 11, 19,
     4, 12 ,20, 5, 13, 21,
     6, 14, 22, 7, 15, 23
   };

   for(i = 0; i < 24; i++)
```

(a) Magic Lens (Bier et al. 1993): Detail of map. Courtesy Xerox Corp.

(b) Cascading overview + detail: SeeSoft (Eick et al. 1992).

(c) Extract and compare: SDM (Roth, Chuah, and Mattis 1995).

(d) Attribute explorer: (Tweedie et al. 1996). Courtesy Robert Spence.

FIGURE 23.24 Interaction techniques.

+ detail displays are thus very helpful for data navigation. Their main disadvantage is that they require coordination of two visual domains.

23.5.4 Linking and Brushing

Overview + detail is an example of coordinating dual representations of the same data. These can be coordinated interactively with *linking and brushing*. Suppose, for example, we wish to show power consumption on an airplane, both in terms of the physical representation of the airplane and a logical circuit diagram. The two views could be shown and *linked* by using the same color for the same component types. Interactivity itself can be used for a dynamic form of linking called *brushing*. In brushing, running the cursor over a part of one of the views causes highlighting both in that view and in the other view.

23.5.5 Extraction and Comparison

We can also use interaction to extract a subset of the data to compare with another subset. An example of this is in the SDM system (Chuah et al. 1995) in Figure 23.24c. The data are displayed in a 3D information landscape, but the perspective interferes with the ability to compare it. Information is therefore *extracted* from the display (leaving ghosts behind) and placed in an orthogonal viewing position where it can be *compared* using 2D. It could also be dropped into another display. Interactivity makes possible these manipulations, while keeping them coordinated with the original representations.

23.5.6 Attribute Explorer

Several of these interactive techniques are combined in the Attribute Explorer (Tweedie et al. 1996). Figure 23.24d shows information on four attributes of houses. Each attribute is displayed by a histogram, where each square making up the histogram represents an individual house. The user selects a range of some attribute, say price. Those pixels making up the histogram on price have their corresponding pixels linked representing houses highlighted on the other attributes. Those houses meeting all the criteria are highlighted in one color; those houses meeting, say, all but one are highlighted in another color. In this way, the user can tell about the "near misses." If the users were to relax one of the criteria only a little (say, reducing price by $100), then the user might be able to gain more on another criterion (say, reducing a commute by 20 miles).

23.6 FOCUS + CONTEXT ATTENTION-REACTIVE ABSTRACTIONS

So far, we have considered visualizations that are static mappings from Data Table to Visual Structure and those where the mappings Data Table to Visual Structure are interactively controlled by the user. We now consider visualizations in which the machine is no longer passive, but its mappings from Visual Structure to View are altered by the computer according to its model of the user's *degree of interest*. We can, in principle, associate a cost of access with every element in the Data Table. Take the FilmFinder in Figure 23.3. Details about the movie *Murder on the Orient Express* are accessible at low cost in terms of time because they are presently visible on the screen. Details of *Goldfinger*, a movie with only a mark on the display, take more time to find. Details of *Last Year at Marienbad*, a movie with no mark on the display, would take much more time. The idea is that with a model for predicting users' changes in interest, the system can adjust its displays to make costs lower for information access. For example, if the user wants some detail about a movie, such as the director, the system can anticipate that the user is more likely to want other details about the movie as well and therefore display them all at the same time. The user does not have to execute a separate command; the cost is therefore reduced.

Focus + context views are based on several premises: First, the user needs both overview (context) and detail information (focus) during information access, and providing these in separate screens or separate displays is likely to cost more in user time. Second, information needed in the overview may be different from that needed in the detail. The overview needs to provide enough information to the user to decide where to examine next or to give a context to the detailed information rather than the detailed information itself. As Furnas (1981) has argued, the user's interest in detail seems to fall away in a systematic way with distance as information objects become farther from current interest. Third, these two types of information can be combined within a single dynamic display, much as human vision uses a two-level focus and context strategy. Information broken into multiple displays (separate legends for a graph, for example) seems to degrade performance due to reasons of visual search and working memory.

Furnas (1981) was the first to articulate these ideas systematically in his theory of *fish-eye views*. The essence of focus + context displays is that the average cost of accessing information is reduced by placing the most likely needed information for navigation and detail where it is fastest to access. This can be accomplished by working on either the data side or the visual side of the visual reference model (Figure 23.10). We now consider these techniques in more detail.

23.6.1 Data-Based Methods

23.6.1.1 Filtering

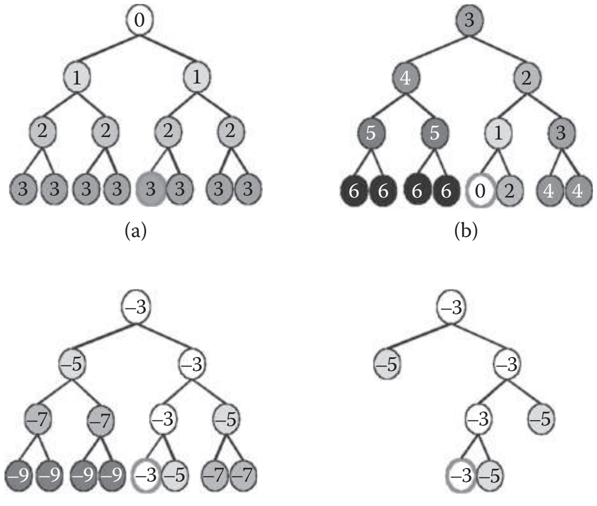
On the data side, focus + context effects can be achieved by filtering out which items from the Data Table are actually displayed on the screen. Suppose we have a tree of categories taken from *Roget's Thesaurus*, and we are interacting with one of these, "Hardness."

Matter
ORGANIC
Vitality
Vitality in general
Specific vitality
Sensation
Sensation in general
Specific sensation
INORGANIC
Solid
Hardness
Softness
Fluid
Fluids in general
Specific fluids

Of course, this is a small example for illustration. A tree representing a program listing or a computer directory or a taxonomy could easily have thousands of lines, a number that would vastly exceed what could fit on the display and hence would have a high cost of accessing. We calculate a degree-of-interest (DOI) for each item of the tree, given that the focus is on the node Hardness. To do this, we split the DOI into an intrinsic part and a part that varies with distance from the current center of interest and use a formula from Furnas (1981).

$$DOI = Intrinsic\ DOI + Distance\ DOI$$

Figure 23.25 shows schematically how to perform this computation for our example. We assume that the intrinsic DOI of a node is just its distance of the root (Figure 23.25a). The distance part of the DOI is just the traversal distance to a node from the current focus node (Figure 23.25b; it turns out to be convenient to use negative numbers for this



(a) (b)

(c) (d)

FIGURE 23.25 Degree-of-Interest calculation for fish-eye visualization.

computation, so that the maximum amount of interest is bounded, but not the minimum amount of interest). We add these two numbers together (Figure 23.25c) to get the DOI of each node in the tree. Then we apply a minimum threshold of interest (–5 in this case) and only show nodes more interesting than that threshold. The result is the reduced tree:

Matter
  INORGANIC
  ORGANIC
    Solid
      Hardness
      Softness
    Fluid

The reduced tree gives local context around the focus node and progressively less detail farther away. But it does seem to give the important context.

23.6.1.2 Selective Aggregation

Another focus1context technique from the data side is selective aggregation. Selective aggregation creates new cases in the Data Table that are aggregates of other cases. For example, in a visualization of voting behavior in a presidential election, voters could be broken down by sex, precinct, income, and party affiliation. As the user drills down on, say, male Democrats earning between $25,000 and $50,000, other categories could be aggregated, providing screen space and contextual reference for the categories of immediate interest.

23.6.2 View-Based Methods

23.6.2.1 Micro-Macro Readings

Micro-macro readings are diagrams in which “detail cumulates into larger coherent structures” (Tufte 1990). The diagram can be graphically read at the level of larger contextual structure or at the detail level. An example is Figure 23.26. The micro reading of this diagram shows three million observations of the sleep (lines), wake (spaces), and feeding (dots) activity of a newborn infant. Each day’s activity is repeated three times on a line to make the cyclical aspect of the activity more clearly visible. The macro reading of the diagram, emphasized by the thick lines, shows the infant transitioning from the natural human 25-hour cycle at birth to the 24-hour solar day. The macro reading serves as context and index into the micro reading.

23.6.2.2 Highlighting

Highlighting is a special form of micro-macro reading in which focal items are made visually distinctive in some way. The overall set of items provides a context for the changing focal elements.

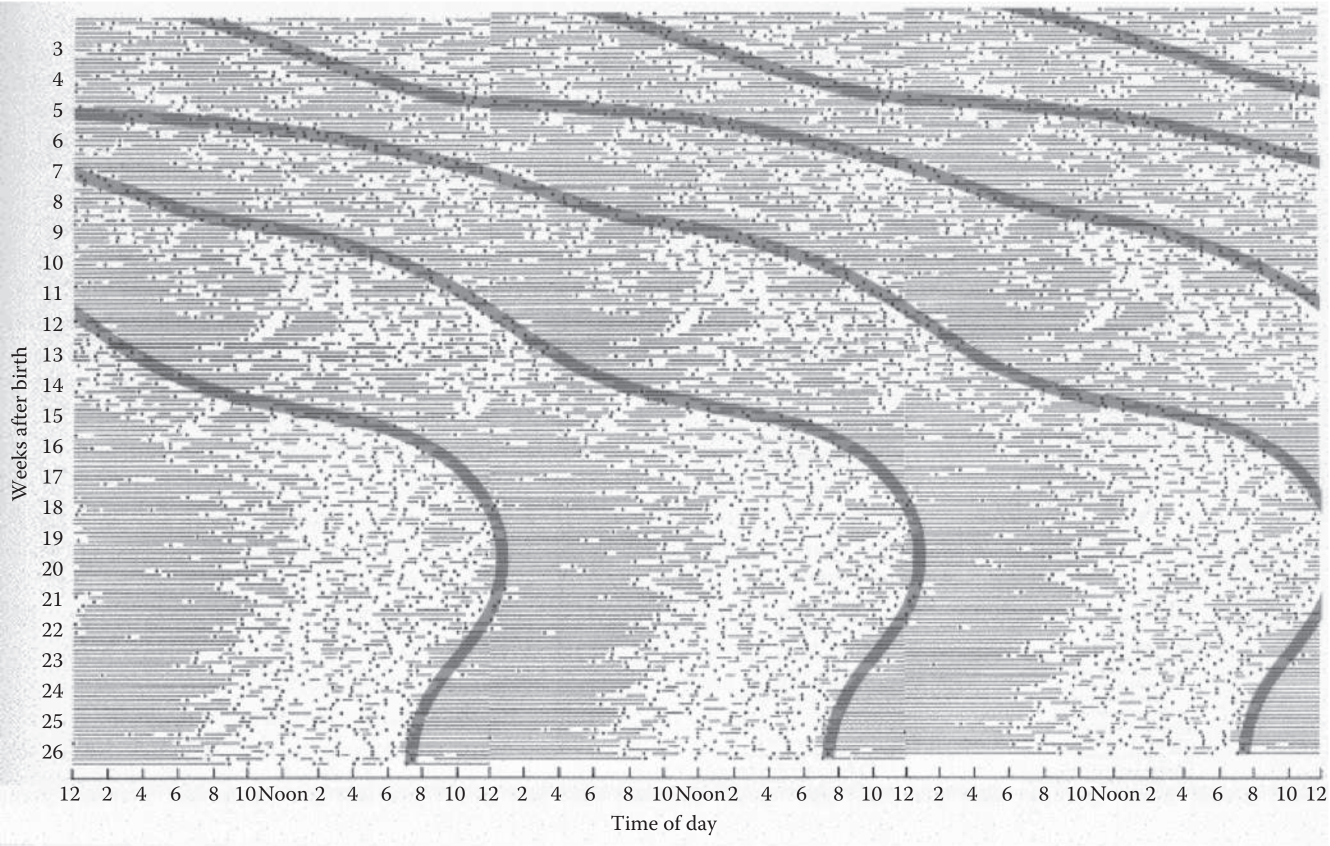


FIGURE 23.26 Micro-macro reading. (Courtesy Scientific American Library.)

23.6.2.3 Visual Transfer Functions

We can also warp the view with viewing transformations. An example is a visualization called the *bifocal lens* (Spence and Apperley 1982). Figure 23.27a shows a set of documents the user would like to view, but which is too large to fit on the screen. In a bifocal lens, documents not in a central focal region are compressed down to a smaller size. This could be a strict visual compression. It could also involve a change in representation. We can talk about the visual compression in terms of a visual transfer function in Figure 23.27b, sometimes conveniently represented in terms of its first derivative in Figure 23.27c. This function shows how many units of an axis in the original display are mapped into how many units in the resultant display. The result could be compression or enlargement of a section of the display. As a result of applying this visual transfer function to Figure 23.27a, the display is compressed to Figure 23.27d. Actually, the documents in the compressed region have been further altered by using a semantic zooming function to give them a simplified visual form. The form of Figure 23.27c shows that this is essentially a step function of two different slopes. An example of a two-dimensional step function is the Table Lens (Figure 23.28a). The Table Lens is a spreadsheet in which the columns of selected cells are expanded to full size in X and the rows of selected cells are expanded to full size in Y. All other cells are compressed, and their content represented only by a graphic. As a consequence, spreadsheets up to a couple orders of magnitude larger can be represented.

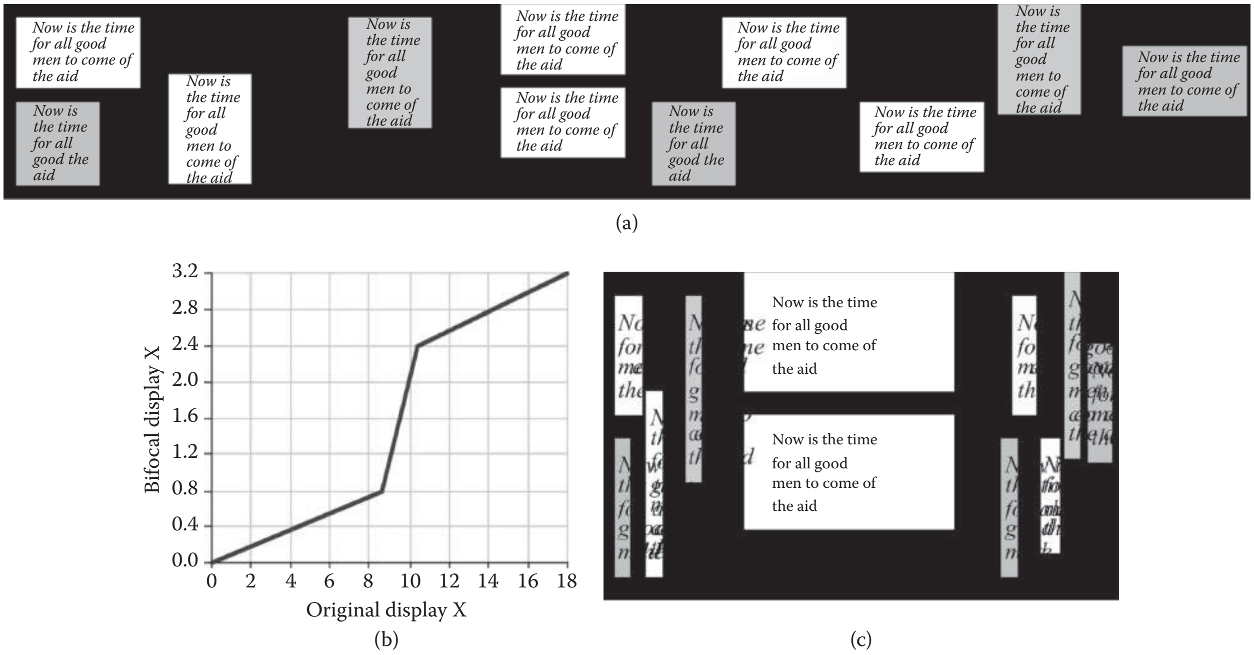
By varying the visual transfer function (see, for example, the review by Leung and Apperley 1994), a wide variety of distorted views can be generated. Figure 23.28b shows an application in which a visual transfer function is used to expand a bubble around a local region on a map. The expanded space in the region is used to show additional information about that region. Distorted views must be designed carefully so as not to damage important visual relationships. Bubble distortions of maps may change whether roads appear parallel to each other. However, distorted views can be designed with “flat” and “transition” regions to address this problem. Figure 23.27a does not have curvilinear distortions. Focus + context visualizations can be used as part of compact user controls. Keahey (2001) has created an interactive scheme in which the bubble is used to “preview” a region. When the user releases a button over the region, the system zooms in far enough to flatten out the bubble. Bederson has developed a focus1context pull-down menu (Bederson 2000) that allows the viewing and selection of large lists of typefaces in text editor, Figure 23.27c.

23.6.2.4 Perspective Distortion

One interesting form of distorting visual transfer functions is 3D perspective. Although it can be described with a 2D distorting visual transfer function, it is usually not experienced as distorting by users due to the special perceptual mechanisms humans have for processing 3D. Figure 23.28c shows the Perspective Wall (Mackinlay, Robertson, and Card 1991). Touching any place on the walls animates its transition into the central focal area. The user perceives the context area of the wall as an undistorted 2D image in a 3D space, rather than as a distorted 2D image; however, the same sort of compression is still achieved in the nonfocus area.

23.6.2.5 Alternate Geometries

Instead of altering the size of components, focus + context effects can also be achieved by changing the geometry of the spatial substrate itself. One example is the hyperbolic tree (Lamping and Rao 1994). A visualization such as a tree is laid out in hyperbolic space (which itself expands exponentially,

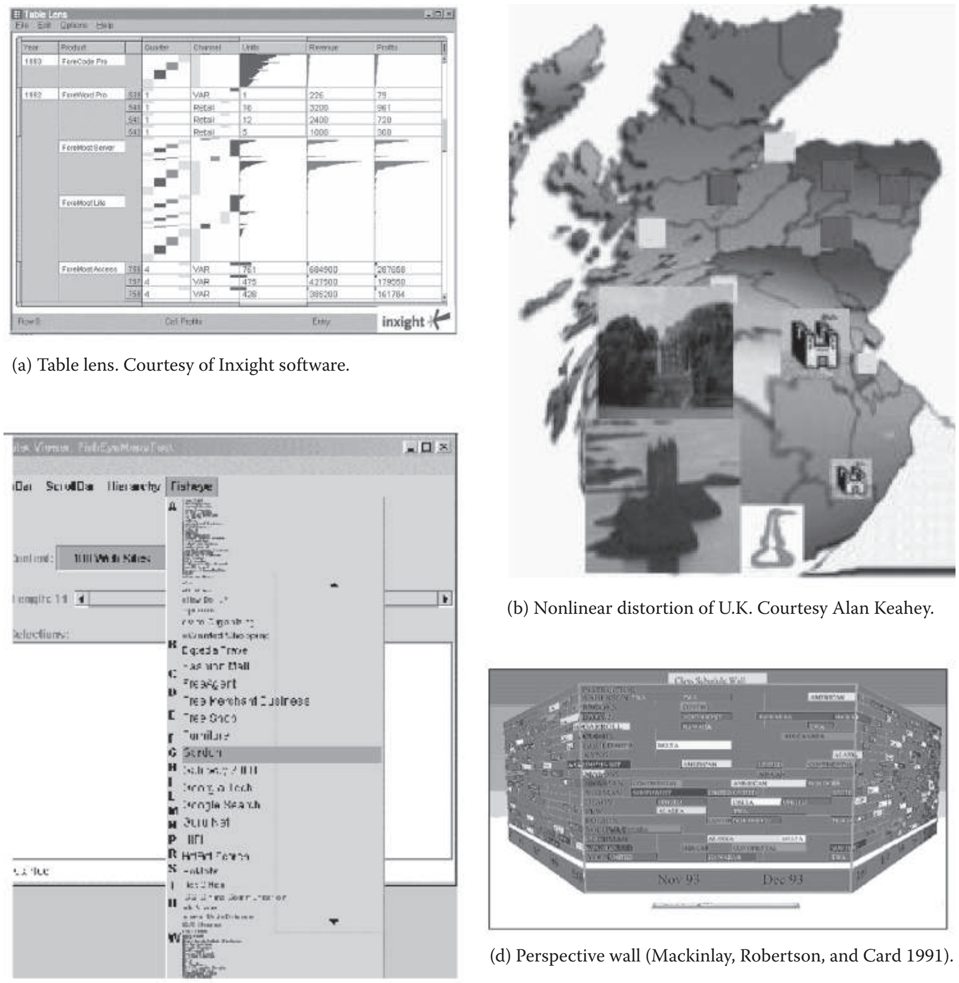


(a)

(b)

(c)

FIGURE 23.27 Bifocal + transfer function.



(a) Table lens. Courtesy of Inxight software.

(b) Nonlinear distortion of U.K. Courtesy Alan Keahey.

(c) Fisheye menus (Bederson 2000).

(d) Perspective wall (Mackinlay, Robertson, and Card 1991).

FIGURE 23.28 Attention-Reactive Visualizations.

just like the tree does), and then projected on to the Euclidean plane. The result is that the tree seems to expand around the focal nodes and to be compressed elsewhere. Selecting another node in the tree animates that portion to the focal area. Munzner (Munzner and Burchard 1995) has extended this notion to 3D hyperbolic trees and used them to visualize portions of the Internet.

23.7 SENSEMAKING WITH VISUALIZATION

23.7.1 Knowledge Crystallization

The purpose of information visualization is to amplify cognitive performance, not just to create interesting pictures. Information visualizations should do for the mind what automobiles do for the feet. So here, we return to the higher level cognitive operations of which information visualization is a means and a component. A recurrent pattern of cognitive activity to which information visualization would be useful (though not the only one!) is "knowledge crystallization." In knowledge crystallization tasks, there is a goal (sometimes ill-structured) that requires the acquisition and making sense of a body of information, as well as the creative formulation of a knowledge product, decision, or action. Examples would be writing a scientific paper, business or military intelligence, weather forecasting, or buying a laptop computer. For these tasks, there is usually a concrete outcome of the task—the submitted manuscript of a paper, a delivered briefing, or a purchase. Knowledge crystallization does have characteristic processes, however, and it is by amplifying these that information visualization seeks to intervene and amplify the user's cognitive powers. Understanding of this process is still tentative, but the basic parts can be outlined:

> Acquire information. Make sense of it. Create something new. Act on it.

In Table 23.10, we have listed some of the more detailed activities these entail. We can see examples of these in our initial examples.

23.7.1.1 Acquire Information

The FilmFinder is concentrated largely on acquiring information about films. *Search* is one of the methods of acquiring

TABLE 23.10
Knowledge Crystallization Operators

Acquire information	Monitor
	Search
	Capture (make implicit knowledge explicit)
Make sense of it	Extract information
	Fuse different sources
	Find schema
	Recode information into schema
Create something new	Organize for creation
	Author
Act on it	Distribute
	Apply
	Act

information in Table 23.10, and the FilmFinder is an instance of the use of information visualization in search. In fact, Shneiderman (Card, Mackinlay, and Shneiderman 1999) has identified a heuristic for designing such systems:

> Overview first, zoom and filter, then details-on-demand

The user starts with an overview of the films, and then uses sliders to filter the movies, causing the overview to zoom in on the remaining films. Popping up a box gives details on the particular films. The user could use this system as part of a knowledge crystallization process, but the other activities would take place outside the system. The SmartMoney system also uses the TreeMap visualization for acquiring information, but this time the system is oriented toward *monitoring,* another of the methods in Table 23.10. A glance at the sort of chart in Figure 23.5 allows an experienced user to notice interesting trends among the hundreds of stocks and industries monitored. Another method of acquiring information, *capture*, refers to acquiring information that is tacit or implicit. For example, when users browse the World Wide Web, their paths contain information about their goals. This information can be captured in logs, analyzed, and visualized (Chi and Card 1999). It is worth making the point that acquiring information is not something that the user must necessarily do explicitly. Search, monitoring, and capture can be implicitly triggered by the system.

23.7.1.2 Make Sense of It

The heart of knowledge crystallization is sensemaking. This process is by no means as mysterious as it might appear. Because sensemaking involving large amounts of information must be externalized, the costs of finding, organizing, and moving information around have a major impact on its effectiveness. The actions of sensemaking itself can be analyzed. One process is *extraction.* Information must be got out of its sources. In our hotel example, the hotel manager extracted information from hotel records. A more subtle issue is that information from different sources must be *fused*—that is, registered in some common correspondence. If there are six called-in reports of traffic accidents, does this mean six different accidents, one accident called in six times, or two accidents reported by multiple callers? If one report merely gives the county, while another just gives the highway, it may not be easy to tell. Sensemaking involves finding some *schema*—that is, some descriptive language—in terms of which information can be compactly expressed (Russell et al. 1993). In our hotel example, permuting the matrices brought patterns to the attention of the manager. These patterns formed a schema she used to organize and represent hotel stays compactly. In the case of buying a laptop computer, the schema may be a table of features by models. Having a common schema then permits compact description. Instances are *recoded* into the schema. Residual information that does not fit the schema is noted and can be used to adjust the schema.

23.7.1.3 Create Something New

Using the schema, information can be reorganized to create something new. It must be *organized* into a form suitable for the output product and that product must be *authored.* In the case of the hotel example, the manager created the presentation of Figure 23.7c.

23.7.1.4 Act on It

Finally, there is some consequential output of the knowledge crystallization task. That action may be to distribute a report or give a briefing, to act directly in some way, such as setting up a new promotion program for the hotel or buying a laptop on the basis of the analysis, or by giving directives to an organization.

23.7.2 Levels For Applying Information Visualization

Information visualization can be applied to facilitate the various subprocesses of knowledge crystallization just described. It can also be applied at different architectural levels in a system. These have been depicted in Figure 23.29. At one level is the use of visualization to help users access information outside the immediate environment—the *infosphere*—such as information on the Internet or from corporate digital libraries. Figure 23.30a shows such a visualization of the Internet (Bray 1996). Websites are laid out in a space such that sites closer to each other in the visualization tend to have more traffic. The size of the disk represents the number of pages in the site. The globe size represents the number of out-links. The globe height shows the number of in-links.

The second level is the *information workspace.* The information workspace is like a desk or workbench. It is a staging area for the integration of information from different sources. An information workspace might contain several visualizations related to one or several tasks. Part of the purpose of an information workspace is to make the cost of access low for information in active use. Figure 23.30b shows a 3D workspace for the Internet, the Web Forager (Card, Robertson, and York 1996). Pages from the World Wide Web, accessed by users through clicking on URLs or searches, appear in the space. These can be organized into piles or books related to different topics. Figure 23.30c shows

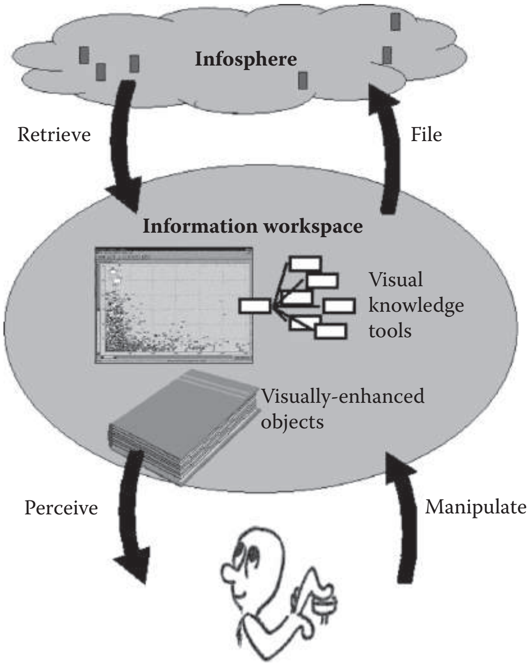


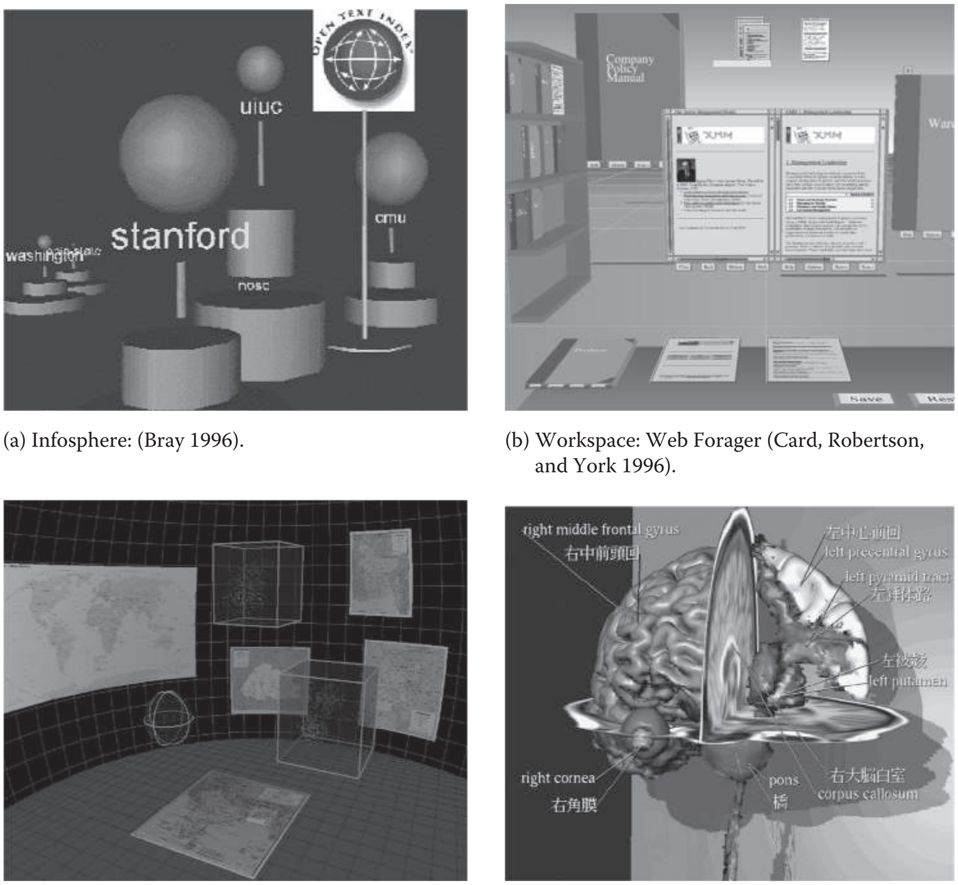
FIGURE 23.29 Levels of use for information visualization.

another document workspace, STARLIGHT (Risch et al. 1997). Documents are represented as galaxies of points in space such that similar documents are near each other. In the workspace, various tools allow linking the documents to maps and other information and analytical resources.

The third level is *visual knowledge tools.* These are tools that allow schema forming and rerepresentation of information. The permutation matrices in Figure 23.7, the SeeSoft system for analyzing software in Figure 23.15b, and the Table Lens in Figure 23.27a are examples of visual knowledge tools. The focus is on determining and extracting the relationships.

The final level is *visually enhanced objects*, coherent information objects enhanced by the addition of information visualization techniques. An example is Figure 23.30d, in which voxel data of the brain have been enhanced through automatic surface rendition, coloring, slicing, and labeling. Abstract data structures representing neural projects and anatomical labels have been integrated into a display of the data. Visually enhanced objects focus on revealing more information from some object of intrinsic visual form.

Information visualization is a set of technologies that use visual computing to amplify human cognition with abstract information. The future of this field will depend on the uses to which it is put and how much advantage it gives to these.



(a) Infosphere: (Bray 1996).

(b) Workspace: Web Forager (Card, Robertson, and York 1996).

(c) Workspace: STARLIGHT (Risch et al. 1997).

(d) Visually-enhanced object: Voxel-Man. Courtesy of University of Hamburg.

FIGURE 23.30 Information visualization applications.

Information visualization promises to help us speed our understanding and action in a world of increasing information volumes. It is a core part of a new technology of human interfaces to networks of devices, data, and documents.

23.8 ACKNOWLEDGMENTS

This chapter benefited from detailed and generous discussion with Jock Mackinlay and tutorial collaborations with Ed Chi of PARC.

REFERENCES

Ahlberg, C., and B. Shneiderman. 1994. Visual information seeking using the FilmFinder. In *Conference companion on Human factors in computing systems, CHI '94*, 433–4. New York: ACM.

Ahlberg, C., and B. Shneiderman 1994b. Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In *Paper Presented at the Proceedings of CHI'94, ACM Conference on Human Factors in Computing Systems*, New York: ACM.

Battista, G. D., P. Eades, R. Tamassia, and I. G. Tollis. 1994. Annotated bibliography on graph drawing. *Comput Geom Theory Appl* 4(5):235–82.

Becker, R. A., S. G. Eick, and A. R. Wilks. 1995. Visualizing network data. *IEEE Trans Vis Comput Graph* 1:16–28.

Bederson, B. B. 2000. *Fisheye Menus Proceedings of ACM Conference on User Interface Software and Technology (UIST 2000)*, 217–26. New York: ACM.

Bertin, J. 1983. *Semiology of Graphics: Diagrams, Networks, Maps* (W. J. Berg, Trans.). Madison, WI: University of Wisconsin Press. (Original work published 1967).

Bertin, J. 1981. Graphics and graphic information-processing. Berlin: Walter de Gruyter.

Bray, T. 1996. Measuring the web. *Comput Netw ISDN Syst* 28(7–11–May):992.

Card, S. K., J. D. Mackinlay, and B. Shneiderman. 1999. *Information Visualization: Using Vision to Think*. San Francisco, CA: Morgan Kaufmann Publishers.

Card, S. K., J. D. Mackinlay, and G. G. Robertson. 1991. The Information Visualizer: An Information Workspace. *ACM Conference on Human Factors in Computing Systems (CHI '91)* 181–8

Card, S. K., T. P. Moran, and A. Newell. 1986. The model human processor: An engineering model of human performance. In *Handbook of Perception and Human Performance*, ed. J. Thomas, 41–35. New York: John Wiley and Sons.

Card, S. K., G. G. Robertson, and W. York. 1996. The webbook and the web forager: An information workspace for the world-wide web. In *Paper Presented at the Proceedings of CHI'96, ACM Conference on Human Factors in Computing Systems*, New York.

Casner, S. 1991. Task-analytic approach to the automated design of graphic presentations. *ACM Trans Graph* 10:111–51.

Chi, E. H., and S. K. Card. 1999. Sensemaking of evolving websites using visualization spreadsheets. In *Paper Presented at the Infovis 1999, IEEE Conference on Information Visualization 1999*, San Francisco.

Chi, E. H.-H., and J. T. Riedl. 1998. An Operator Interaction Framework for Visualization Systems. In *IEEE Symposium on Information Visualization,* North Carolina, 19–20. October 1998, IEEE CS, 63–70.

Chuah, M. C., S. F. Roth, J. Mattis, and J. A. Kolojejchick. 1995. Sdm: Malleable information graphics. In *Paper presented at the Proceedings of InfoVis'95, IEEE Symposium on Information Visualization*, New York.

Cleveland, W. S., and McGill, M. E. (1988). *Dynamic Graphics for Statistics*. Pacific Grove, California: Wadsworth and Brooks/Cole.

Cruz, I. F., and R. Tamassia. 1998. Graph drawing tutorial. http://www.cs.brown.edu/~rt/papers/gd-tutorial/gd-constraints.pdf(accessed on October 25, 2011).

Eick, S. G., J. L. Steffen, and E. E. Sumner. 1992. Seesoft—a tool for visualizing software. *IEEE Trans Softw Eng* 18(11):957–68.

Eick, S. G., and G. J. Wills. 1993. Navigating large networks with hierarchies. In *Paper Presented at the Proceedings of IEEE Visualization'93 Conference*, San Jose, CA.

Engelhardt, Y., J. De Bruin, T. Janssen, and R. Scha. 1996. The visual grammar of information graphics, artificial intelligence in design. Artificial Intelligence in Design (AID '96) in the Workshop on Visual Representation, Reasoning and Interaction in Design 24–27 of June, Stanford, CA.

Fairchild, K. M., S. E. Poltrock, and G. W. Furnas. 1988. Semnet: Three-dimensional representations of large knowledge bases. In *Cognitive science and its applications for human-computer interaction*, ed. R. Guindon, 201. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Feiner, S. K., and C. Beshers. 1990. Worlds within worlds: metaphors for exploring n-dimensional virtual worlds. In *Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology, UIST '90*, 76–83. New York, NY: ACM.

Freeman, E., and S. Fertig, "Lifestreams: Organizing your electronic life," 1995. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.48.6769

Furnas, G. W. 1981. The fisheye view: A new look at structured files. In *Readings in Information Visualization: Using Vision to Think*, ed. B. Shneiderman, 312–30. San Francisco, CA: Morgan Kaufmann Publishers, Inc.

Healey, C. G., K. S. Booth, and J. T. Enns. 1995. High-speed visual estimation using preattentive processing. *ACM Trans Comput-Hum Interact* 3(2):107–35.

Hutchins, E. L. 1996. *Cognition in the wild*. Cambridge, MA: MIT Press.

Inselberg, A. 1997. Multidimensional detective. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*. Washington, DC: IEEE Computer Society.

Inselberg, A., and B. Dimsdale. 1990. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *Paper Presented at the Proceedings of IEEE Visualization'90 Conference*, Los Alamitos, CA.

Keahey, T. A. 2001. Getting along: Composition of visualization paradigms. In *Paper Presented at the Infovis 2001, IEEE Information Visualization 2001*, San Diego, California.

Keim, D. A., and H.-P. Kriegel. 1994. Visdb: Database exploration using multidimensional visualization. *IEEE Comput Graph Appl* 14:40–9.

Kosslyn, S. M. 1994. *Image and Brain: The Resolution of the Imagery Debate*. Cambridge, MA: The MIT Press.

Lamping, J., and R. Rao. 1994. Laying out and visualizing large trees using a hyperbolic space. *Proceedings of the 7th annual ACM symposium on User interface software and technology UIST 94*, 1(November), 13–4. ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=192426.192430

Larkin, J. H., and Simon, H. A. 1987. Why a diagram is (sometimes) worth 10,000 word. *Cognitive Science* 65–99.

Leung, Y. K., and M. D. Apperley. 1994. A review and taxonomy of distortion-orientation presentation techniques. *ACM Trans Comput-Hum Interact* 1(2):126–60.

Lin, X., D. Soergel, and G. Marchionini. 1991. A self-organizing semantic map for information retrieval. In *Paper Presented at the Proceedings of SIGIR'91, ACM Conference on Research and Development in Information Retrieval*, Chicago, IL.

MacEachren, A. M. 1995. *How Maps Work*. New York: The Guilford Press.

Mackinlay, J. D. 1986a. *Automatic Design of Graphical Presentations*. Unpublished doctoral dissertation, Stanford University, California.

Mackinlay, J. D. 1986b. Automating the design of graphical presentations of relational information. *ACM Trans Graph* 5(2):110–41.

Mackinlay, J. D., G. G. Robertson, and S. K. Card. 1991. The perspective wall: Detail and context smoothly integrated. In *Paper Presented at the Proceedings of CHI'91, ACM Conference on Human Factors in Computing Systems*, New York.

McCormick, B. H., and T. A. DeFanti. 1987. Visualization is scientific computing. *Comput Graph* 21:15–21.

Munzner T., Burchard P. 1995. Visualizing the Structure of the World Wide Web in 3D Hyperbolic Space, Special *issue of Computer Graphics,* ACM SIGGRAPH, 33–8. New York: ACM.

Norman, Donald A. 1993. Things That Make Us Smart: Defending Human Attributes in the Age of the Machine. Reading, MA: Perseus.

Playfair, W. 1786. *The Commercial and Political Atlas*. London.

Resnikoff, H. L. 1989. *The Illusion of Reality*. New York: Springer-Verlag.

Risch, J. S., D. B. Rex, S. T. Dowson, T. B. Walters, R. A. May, and B. D. Moon. 1997. The starlight information visualization system. In *Paper Presented at the Proceedings of IEEE International Conference on Information Visualization*, London, England.

Robertson, G. G., S. K. Card, and J. D. Mackinlay. Mackinlay. 1989. The cognitive co-processor for interactive user interfaces. In *Paper Presented at the Proceedings of UIST'89, ACM Symposium on User Interface Software and Technology*, 10–8. New York: ACM.

Robertson, G. G., S. K. Card, and J. D. Mackinlay. 1993. Information visualization using 3d interactive animation. *Commun ACM* 36(4):57–71.

Robin, H. 1992. *The Scientific Image: From Cave to Computer*. New York: H. N. Abrams, Inc.

Roth, S. F., and J. Mattis. 1990. Data characterization for intelligent graphics presentation. In *Paper Presented at the Proceedings of CHI'90, ACM Conference on Human Factors in Computing Systems*, New York.

Russell, D. M., M. J. Stefik, P. Pirolli, and S. K. Card. 1993. The cost structure of sensemaking. In *Paper Presented at the Proceedings of INTERCHI'93, ACM Conference on Human Factors in Computing Systems*, Amsterdam.

Shneiderman, B. 1992. Tree visualization with tree-maps: A 2-dimensional space filling approach. *ACM Trans Graph*, 11(1):92–9.

Shneiderman, B., and M. Wattenberg. 2001. Ordered tree layouts. In *Paper Presented at the IEEE Symposium on Information Visualization, San Diego*, California.

Spence, R. 2000. *Information Visualization*. Harlow, England: Addison-Wesley.

Spence, R., and M. Apperley. 1982. Data base navigation: An office environment for the professional. *Behav Inf Technol* 1(1):43–54.

Tamassia, R. 1996. Strategic directions in computational geometry working group report. *ACM Comput Surv* 28:591–606.

Tufte, E. R. 1983. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.

Tufte, E. R. 1990. *Envisioning Information*. Cheshire, CT: Graphics Press.

Tufte, E. R. 1997. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics Press.

Tukey, J. W. 1977. *Exploratory Data Analysis*. Reading, MA: Addison-Wesley.

Tweedie, L., R. Spence, H. Dawkes, and H. Su, 1996. Externalising abstract mathematical models. In *Proceedings CHI'96*, 406–12. New York: ACM Press.

Ware, C. 2000. *Information Visualization: Perception for Design*. San Francisco: Morgan Kaufmann Publishers.

24 Collaboration Technologies

Gary M. Olson and Judith S. Olson

CONTENTS

24.1 INTRODUCTION

Computing and communication technologies have provided us with useful and powerful information resources, remote instruments, and tools for interacting with each other. These possibilities have also led to numerous social and organizational effects. These tools are of course just the latest in a long line of modern technologies that have changed human experience. Television and radio long ago broadened our awareness of and interest in activities all over the world. The telegraph and telephone enabled new forms of organization to emerge. The new technologies of computer-supported cooperative work (CSCW) are giving us greater geographical and temporal flexibility in carrying out our activities. They have also given us new modes of socializing.

In this chapter, we will review software designed to run over a network in support of the activities of a group or organization. These activities can occupy any of several combinations of same/different places and same/different times. Software has been designed for all four of these combinations. Early applications tended to focus on only one of these cells, but more recently, software that supports several cells and the transitions among them has emerged. These technologies support collaborative activities at many levels of social aggregation. Both the individual members of groups and the organizations in which they are embedded affect and are affected by collaborative technologies.

A brief note on terminology: in the several decades since networked computing made possible the kinds of software functions we review in this chapter, terms have changed. It was quite popular in the 1990s, for example, to refer to such software as “groupware.” However, as Grudin and Poltrock (2011) point out in their excellent historical review of trends in CSCW, this term has largely been supplanted by terms more neutral as to the level of social aggregation involved. Hence, in this chapter we will use the term collaboration technology. All of the technologies we review have some bearing on how people collaborate with each other. And, again as Grudin and Poltrock point out, the field has moved beyond the focus on work only (see also Crabtree, Rodden, and Benford [2005]), though in some quarters this has been lamented (Schmidt [2010] sets forth the view that the focus on work is significant enough to merit an undiluted field of study).

CSCW emerged as a formal field of study in the mid-1980s, with conferences, journals, books, and university courses appearing that used this name. There were a number of important antecedents. The earliest efforts to create groupware used time-shared systems but were closely linked to the development of key ideas that propelled the personal computer revolution. Bush (1945) described a vision of something similar to today's World Wide Web in an influential essay published shortly after the end of World War II. Doug Engelbart's famous demonstration at the 1968 International Federation of Information Processing Societies meeting in San Francisco included a number of key groupware components (see Engelbart and English [1968]). These components included support for real-time face-to-face (FTF) meetings, audio and video conferencing, discussion databases, information repositories, and workflow support. Group decision-support systems and computer-supported meeting rooms were explored in a number of business schools (see McLeod [1992]; Kraemer and Pinsonneault [1990]). Work on office automation included many groupware elements, such as group workflow management, calendaring, e-mail, and document sharing (Ellis and Nutt 1980). A good summary of early historical trends as well as reprints of key early articles appear in three early anthologies: Greif (1988), Marca and Bock (1992), and Baecker (1993).

Today, there are a large number of commercial collaboration products. In addition, collaboration functions are now appearing as options in operating systems or specific applications (e.g., Windows, Mac, and Linux operating systems, suites of tools by Microsoft, Google, Yahoo!, and many others). Collaborative functionality has become widespread and familiar. However, there are still many research issues about how to design such systems and what effects they have on the individuals, groups, and organizations that use them.

Let us clarify what this chapter is about. It is not a general review of the field of CSCW—that would be an enormously larger task than we can take on here. Rather, we are going to focus on the kinds of collaborative applications that have emerged and achieved wide adoption. We will describe their characteristics, in terms of what functions they serve, and we will mention *some* of the studies done to evaluate them. These studies are a mix of controlled laboratory studies and studies of the technologies in real organizations. For many of the technologies we review, entire chapters could be written about the work that has been done with them. So we will, of necessity, have to be selective in our coverage. Our goal is to be *representative*, not exhaustive.

24.2 ADOPTING GROUPWARE IN CONTEXT

Collaborative systems are often intended to support groups, which are usually embedded in an organization. As a result, there are a number of issues that bear on the success of such systems. In a justly famous set of papers, Grudin (1988, 1994) pointed out a number of problems that such systems have (see also Markus and Connolly [1990]). In brief, he pointed out that developers of such systems need to be concerned with the following issues (Grudin 1994, p. 97; we here use the groupware terminology that he used in these original articles):

1. Disparity in work and benefit. Groupware applications often require additional work from individuals who do not perceive a direct benefit from the use of the application.
2. Critical mass and Prisoner's dilemma problems. Groupware may not enlist the "critical mass" of users required to be useful, or can fail because it is never in any one individual's advantage to use it.
3. Disruption of social processes. Groupware can lead to activity that violates social taboos, threatens existing political structures, or otherwise de-motivates users crucial to its success.
4. Exception handling. Groupware may not accommodate the wide range of exception handling and improvisation that characterizes much group activity.
5. Unobtrusive accessibility. Features that support group processes are used relatively infrequently, requiring unobtrusive accessibility and integration with more heavily used features.
6. Difficulty of evaluation. The almost insurmountable obstacles to meaningful, generalizable analysis and evaluation of groupware prevent us from learning from experience.
7. Failure of intuition. Intuitions in product development environments are especially poor for multiuser applications, resulting in bad management decisions and error-prone design processes.
8. The adoption process. Groupware requires more careful implementation (introduction) in the workplace than product developers have confronted.

There are reasons, however, for optimism. One specific example is Palen and Grudin's (2002) follow-up study of the adoption of group calendars. They found that organizational conditions in the 1990s were much more favorable for the adoption of group tools than they were in the 1980s. Further, the tools themselves had improved in reliability, functionality, and usability. There is increased "collaboration readiness" and "collaboration technology readiness" (Olson and Olson 2000) that has made for increased success of such applications. Indeed, the very rapid take-up of technologies like Wikipedia, Facebook, Twitter, and multiplayer games (and many other examples) is testament to the changed circumstances in current times. Certainly items 5 and 6 in Grudin's list are not as much of a challenge at present. But many of the others persist as serious challenges.

24.3 TECHNICAL INFRASTRUCTURE

Collaborative technology requires networks, and network infrastructure is a key enabler as well as a constraint on such systems. High-quality broadband networking has emerged across most parts of the developed world. Good access to the Internet is now common in many places other than the office, such as homes, hotels, coffee shops, airports, and even in many open spaces like parks. It is also the case that networking infrastructure is spreading throughout the world. However, heterogeneity in network conditions remains a major technical challenge. For instance, doing web conferencing when some participants are on slow dial-up lines and others are on fast advanced networks requires special coordination. Some good resources on the latest developments in networking are Comer (2008) and Kurose and Ross (2009).

The World Wide Web and its associated tools and standards have had a major impact on the possibilities for collaboration (Schatz and Hardin 1994; Berners-Lee 1999). Early collaboration technologies mostly consisted of standalone applications that had to be downloaded and run on each client machine. Increasingly, collaborative tools are written for the web, requiring only a web browser and perhaps some plug-ins. This makes it much easier for the user and also helps with matters such as version control. It also enables better interoperability across hardware and operating systems. The emergence of Web 2.0 has helped create a plethora of interesting applications in a wide variety of areas. For example, many conferencing tools are now accessed through a web browser. See Bell (2009) and Campasoto and Nilson (2011) for a variety of examples and details.

Another major technical advancement has been the explosion of collaborative functions on mobile devices. Laptops, personal digital assistants, wearables, pads, and cell phones provide access to information and people from almost anywhere. More and more applications are being written to operate across these diverse environments (e.g., Tang et al. 2001; Starner and Rhodes 2004; Wiltse and Nichols 2009; Gunaratne and Brush 2010). These devices vary in computational power, display size and characteristics, network bandwidth, and connection reliability, providing interesting technical challenges to make them all interoperate smoothly. For instance, accessing websites from a cell phone requires special user interface methods to make the tiny displays usable (Jones and Marsden 2006). More information about these advances is available in Ling and Donner (2009), and an interesting analysis of the implications of these advances in mobile communication is in Rheingold (2002).

Security on the Internet continues to be a major challenge for collaboration technologies. In some sense, the design of Internet protocols are to blame, since the Internet grew up in a culture of openness and sharing (Longstaff et al. 1997; Abbate 1999; Tanenbaum 2011). E-commerce and sensitive application domains like medicine have been a driver for advances in security, but there is still much progress to be made (Longstaff et al. 1997; Camp 2000). Coping with firewalls that block access to certain organizations can limit the flexibility of web conferencing. A good recent discussion of these issues is in Wong and Yeung (2009).

Additional flexibility is being provided by the development of infrastructure that lies between the network itself and the applications that run on client workstations, called "middleware." This infrastructure makes it easier to link together diverse resources to accomplish collaborative goals. For instance, the emerging Grid technologies allow the marshalling of powerful, scattered computational resources (Foster and Kesselman 2004). Middleware provides such services as identification, authentication, authorization, directories, and security in uniform ways that facilitate the interoperability of diverse applications. All of these technical elements are components of cyber infrastructure (Atkins et al. 2003). There is considerable interest in the development of this infrastructure because of its large impact on research, education, and commerce.

24.3.1 Communication Tools

We now turn to a review of specific kinds of collaboration technologies, highlighting their various properties and uses. We have grouped this review under several broad headings. We do not aim to be exhaustive, but rather seek to illustrate the variety of kinds of tools that have emerged to support human collaborative activities over networked systems. We also highlight various research issues pertaining to these tools.

24.3.2 E-mail

E-mail has become a ubiquitous communication tool. The early adoption of standards made it possible for messages to be exchanged across networks and different base machines and software applications. E-mail is now also done from cell phones, personal digital assistants (PDAs), television sets, and kiosks in public sites. Documents of many types can be easily exchanged. Because of its widespread use, it has often been called the first successful collaboration technology (Sproull and Kiesler 1991; Satzinger and Olfman 1992; O'Hara-Devereaux and Johnson 1994; Anderson et al. 1995). Indeed, it has become so successful that e-mail overload has become a major problem (Whittaker and Sidner 1996). And of course, it has become a vector for viruses, worms, and other malware.

Researchers have shown that this widespread use has had a number of effects on how people behave. It has had large effects on communication in organizations: it changes the social network of who talks to whom (Sproull and Kiesler 1991; DeSanctis, et al. 1996), the power of people who formerly had little voice in decisions (Finholt, Sproull, and Kiesler 1990), and the tone of what is said and how it is interpreted (Sproull and Kiesler 1991). For example, with e-mail, people who were shy found a voice; they could overcome their reluctance to speak to other people by composing text, not speech to another face. This invisibility, however, also has a more general effect—without the social cues in the recipient's face being visible to the sender, people will "flame," send harsh or extremely emotive (usually negative) messages (Arrow et al. 1996; Hollingshead, McGrath, and O'Connor 1993).

As with a number of other "designed" technologies, people use e-mail for things other than the original intent. People use it for managing time, reminding them of things to do, and keeping track of steps in a workflow (Mackay 1989; Carley and Wendt 1991; Whittaker and Sidner 1996). But because e-mail was not designed to support these tasks, it does not do it very well; people struggle with reading signals about whether they have replied or not (and to whom it was cc'd); they manage folders poorly for reminding them to do things, and so forth.

In addition, because e-mail is so widespread, and it is easy and free to distribute a single message to many people, people experience information overload. Many people get hundreds of e-mail messages each day, many of them mere broadcasts of things for sale or events about to happen, much like "classifieds" in the newspaper. Several early efforts to use artificial intelligence techniques to block and/or sort incoming e-mail were tried, and this has continued to be a very active area of work (Malone et al. 1989; Winograd 1988). There are two broad classes of uses of e-mail filters. One use is to automatically sort incoming mail into useful categories. This is relatively easy for mail that has simple properties, such as a person's name. It is more difficult for subtle properties. The other major use is to weed out unwanted mail, such as spam. The state-of-the-art in spam filtering was in the range of 80%–90% effectiveness in 2005 (e.g., Federal Trade Commission 2005). Such filters are so good that many institutions automatically filter mail as it comes in to the organization's gateway, sparing users the need to do it in their own clients. Similarly, many clients now come with built-in spam filters that can be tuned by the user (e.g., Google's Gmail).

These problems have led to the "reinvention" of e-mail (Whittaker, Bellotti, and Moody 2005). For example, given that e-mail is often used in the context of managing projects, systems have been explored that have a more explicit scheme for task management (Whittaker 2005; Bellotti et al. 2005). To deal with problems of e-mail overload, new schemes for filtering e-mail have been explored, such as routing messages differently to different kinds of clients (e.g., cell phone vs. desktop machine; see Schmandt and Marti [2005]). Another approach has been to explore pricing mechanisms for e-mail that are analogous to pricing for regular mail (Kraut et al. 2005). In such schemes one would pay to send e-mail, with higher prices presumably indicating higher priority, analogous to the difference between first class postage and bulk rates. These schemes are exploratory, but are likely to result in new options in future e-mail clients.

Kraut et al. (1998) reported that greater Internet use, which in their sample was mostly e-mail, led to declines in social interactions with family members and an increase in depression and loneliness. Not surprising, these results triggered widespread discussion and debate, both over the substance of the results and the methods used to obtain them. Kraut, Gergle, and Fussell (2002) reported new results that suggested these initial negative effects may not persist. Interpersonal communication is one of the principal uses of the Internet, and the possible implications of this kind of communication for social life are important to understand (see reviews by Bargh and McKenna [2004] and Benkler [2006]). Indeed, Putnam (2000) has wondered whether the Internet can be a source of social cohesiveness. These kinds of questions need to be addressed by additional large-scale studies of the kind carried out by Kraut and his colleagues (see review by Resnick [2002]).

24.3.3 Conferencing Tools: Voice, Video, Text

There are many options available today for on-line conferencing among geographically dispersed members of a group. So-called computer-mediated communication (CMC) has become widespread. There are three principal modes of interaction, but each has numerous subtypes:

- Video + Audio
  - Full-scale video conferencing room; many options for specific design
  - Individual desktop video; many options for quality, interface
  - Conferencing options on mobile devices
- Audio
  - Phone conference
  - Voice over IP (Skype being a very popular application)
- Text
  - Instant messaging, chat, SMS on mobile phones

CMC was an early research focus for CSCW, and much of what we know dates from the early studies. To be sure, there have been some recent refinements of this literature, for which we will mention a few examples.

There are many studies that compare FTF with various forms of CMC. There are some clear generalizations from such work. The main one is that CMC is more difficult to do than FTF and requires more preparation and care (Hollingshead, McGrath, and O'Connor 1993; McLeod 1992; Olson, Olson, and Meader 1995; Siegel et al. 1986; Straus 1996, 1997; Straus and McGrath 1994). A variety of things that come for free in FTF are either difficult to support or outright missing in CMC (Kiesler and Cummings 2002). Backchannel communication, which is important for modulating conversation, is either weak or nonexistent in CMC, although it has become common to keep an instant messaging (IM) chat going during audio or video conferences (e.g., Kellogg et al. 2006). Paralinguistic cues that can soften communication are often missing. Participants in CMC tend to have an informational focus, which means there is usually less socializing, less small talk. Over time, this can lead to poorer social integration and organizational effectiveness (Nohria and Eccles 1992).

CMC often introduces delay. This is well known to be very disruptive to communication (Egido 1988; Krauss and Bricker 1966; O'Conaill, Whittaker, and Wilbur 1993; Ruhleder and Jordan 2001; Tang and Isaacs 1993). Participants will communicate less information, be more frustrated with the communication, and actually terminate communication sessions sooner. Delay can be managed, but it takes special care among the participants and turn-taking widgets in the interface of the tools being used. For instance, if there is delay, then full-duplex open communication will not work, since participants will step all over each other's communication. Either the participants must use a social protocol (e.g., like that used in radio communications with spacecraft), or they must employ a mike-passing procedure with interface indications of who wants to talk next.

Although it might seem desirable to always have the maximum communication and tool support possible, it is not always possible or even necessary to do so. Research shows that effective real-time collaboration can take place under a number of different arrangements, depending on the task, the characteristics of the participants, the specific geographical dispersion of the participants, and the processes employed to manage the interactions. There are also organizational effects, especially when the real-time collaborations are embedded in ongoing activities, as they almost always are.

For instance, early work (Williams 1977) showed that, in referential communication tasks, full-duplex audio is just as effective as FTF. Subsequent research comparing audio and video conferencing (see summaries in Finn, Sellen, and Wilbur [1997]; Cadiz et al. [2000] found similar results for a tutored video-instruction task) showed that for many tasks audio is sufficient and that video adds nothing to task effectiveness, though participants usually report they are more satisfied with video. There are important exceptions, however. Negotiation tasks are more effective with video (Short, Williams, and Christie 1976). This is probably because the more subtle visual cues to the participants' intentions are important in this kind of task. Further, Veinott et al. (1999) found that when participants have less common ground, video helps. In their case, participants were nonnative speakers of English who were doing the task in English. For native speakers, video was no better than audio, but nonnative speakers did better when they had video. Again, visual cues to comprehension and meaning likely played an important role. Recently, an experimental study by Daly-Jones, Monk, and Watts (1998) showed that high-quality video resulted in greater conversational fluency over just high-quality audio, especially as group size increased. There was also a higher rated sense of presence in the video conditions.

An important lesson to draw from this literature is that there are two broad classes by which we might assess whether video is important in real-time collaboration. On the one hand, except for tasks like negotiation or achieving common ground, groups are able to get their work done effectively with high-quality audio. However, for things like satisfaction, conversational fluency, and a sense of presence, video adds value. These kinds of factors might be very important for long-term organizational consequences like employee satisfaction. As of yet, no long-term studies have been done to examine this conjecture.

Audio quality is critical. Ever since early literature review (Egido 1988), it has been reported over and over again that if the audio is of poor quality participants will develop a workaround. For instance, if the audio in a video conferencing system or in a web conferencing system is poor quality, participants will turn to a phone conference.

The social ergonomics of audio and video are also keys to their success. Many of the failures of audio conferencing, especially over the Internet, result from poor-quality microphones, poor microphone placement, poor speakers, and interfering noises like air conditioning. Getting these details right is essential. Similarly, for video, camera placement can matter a lot. For instance, Huang, Olson, and Olson (2002) found that a camera angle that makes a person seem tall (as opposed to actually being tall) affects how influential a person is in a negotiation task. Apparent height matters a lot. Other aspects of camera placement or arrangement of video displays make a big difference as well but are not well known.

An exception is eye contact or gaze awareness, where studies of FTF communication show that these are key linguistic and social mediators of communication (Argyle and Cook 1976; Kendon 1967). It is very difficult to achieve eye contact in CMC systems. Many attempts have been made (Gale and Monk 2000; Grayson and Monk 2003; Monk and Gale 2002; Okada et al. 1994; Vertegaal 1999; Vertegaal et al. 2001), and at least the subjective reports are that these can be effective. But these all require special equipment or setups. And they do not scale very well to multiparty sessions. A recent study by Nguyen and Canny (2007) showed that a relatively simple and inexpensive setup that supports gaze awareness (being able to tell who is looking at whom) had a clear effect on a task that assessed the formation of interpersonal trust.

While for most situations having at least high-quality audio is essential, there are some special cases where a text-based channel, like chat or instant messaging, can work fine. For instance, in the Upper Atmospheric Research Collaboratory (UARC, later known as the "Space Physics and Aeronomy Research Collaboratory" or SPARC), a chat system worked very well for carrying out geographically distributed observational campaigns, since the flow of events in these campaigns were relatively slow (campaigns went on for several days, key events would take many minutes to unfold). McDaniel, Olson, and Magee (1996) compared chat logs with earlier FTF conversations at a remote site and found many elements of them very similar, including informal socializing. But this kind of ongoing scientific campaign is very unlike the interactions that take place in a typical meeting.

Instant messaging is a new communication modality that is making substantial inroads into organizations. Muller et al. (2003) found in a survey study of three organizations that the introduction of instant messaging led to significantly less use of such communication channels as e-mail, voice-mail, telephone, teleconference, pager, and FTF. They also found that instant messaging was used for "substantive business purposes." Furthermore, in one of the organizations where they surveyed users after 24 months of usage they found that the substantive reasons for using IM increased. In a study of IM logs in an organization, Isaacs et al. (2002) found that a large proportion of IM conversations involved "complex work discussions." They found that IM users seldom switched to another communication channel once they were engaged in

IM. Nardi, Whittaker, and Bradner (2000) observed in a field study that workers used IM for a variety of purposes, not just for information exchange. Such matters as quick questions, scheduling, organizing social interactions, and keeping in touch with others were common uses of IM. Thus, IM has emerged as a significant communication medium in the workplace and is used even when other, richer communication channels were available.

Although IM is a relatively new phenomenon in the workplace, it is clearly established as a useful and widely used tool outside the workplace. This will undoubtedly assist in the development of more sophisticated versions of the tool, as well as its integration into on-line conferencing systems. There is clearly much promise here. We have noticed, for example, that during online conferences IM or chat serves as a backchannel for side conversations or debugging, an extremely useful adjunct to the core audio or video communication taking place in such conferences.

The other key feature of successful remote meetings is the ability to share the objects they are talking about, such as the agenda, the to-do list, the latest draft of a proposal, a view of an object to be repaired, and so on. Many researchers (Fussell, Kraut, and Siegel 2000; Karsenty 1999; Kraut, Fussell, and Siegel 2003; Kraut et al. 2002; Luff et al. 2003; Nardi et al. 1993; Whittaker, Geelhoed, and Robinson 1993) have provided experimental evidence of the value of a shared workspace for synchronous audio-supported collaboration. More traditional video conferencing technologies often offer an "object camera," onto which the participants can put a paper agenda, Powerpoint slides, or a manufactured part. More generally, any form of video can also be used to share work objects (Fussell, Kraut, and Siegel 2000; Nardi et al. 1993). For digital objects, there are now a number of products that will allow meeting participants to share the screen or, in some cases, the remote operation of an application. Some companies are using electronic whiteboards, both in a collocated meeting and in remote meetings to mimic the choreography of people using a physical whiteboard. In some "collaboratories," scientists can even operate remote physical instruments from a distance and jointly discuss the results.

There are a growing number of studies that have looked at cultural issues in CMC. For example, Setlock, Fussell, and Neuwirth (2004) studied the differences in conversational content between Chinese and American pairs while engaged in a decision-making task, either in a FTF situation or using instant messaging, and found a series of differences in how they conversed. On the basis of analyses of agreement and efficiency, along with some specific text analyses, they characterized pairs of Americans as viewing the task as one of working out a mutually acceptable joint rating, whereas the Chinese pairs worked to reach agreement on the relative worth of the specific items to be rated. Fussell and her colleagues have carried out a series of studies comparing CMC behaviors across Asian and American cultures (e.g., Diamant, Fussell, and Lo 2009; Wang, Fussell, and Setlock 2009). A recent review of work on multicultural teams is by Connaughton and Shuffler (2007). The emergence of a new conference series on Intercultural Collaboration (e.g., International Conference on Intercultural Collaboration or ICIC 2010) provides a focused venue for work like this.

24.3.4 Blogs

Weblogs, or more commonly called "blogs," have burst upon the Internet scene in recent years. Blogging software that makes it easy to put up multimedia content has led people to set up sites for all manner of purposes. A site can contain text, pictures, movies, and audio clips. A common social purpose is to keep an on-line diary. Another is to provide commentary on a topic of interest. For instance, blogs played a major role in the 2004 election (Adamic and Glance 2005). Nardi et al. (2004) studied why people blog, as it is sometimes puzzling that people would essentially share personal or private information about themselves through the web. Blogging has emerged as a major research topic in this area, and the literature is growing apace.

A special topic related to blogging is the emergence of microblogging, best instantiated by Twitter. Twitter limits contributions to 140 characters, so of necessity "tweets," as they are called, are concise. Yet Twitter has emerged as a major social phenomenon, and of course is also receiving significant scholarly attention. A couple of recent examples of such studies include Huberman, Romero, and Wu (2009) and Zhao and Rosson (2009).

24.3.5 Disaster Response

Within the past decade the kinds of CMC tools we have been reviewing have emerged as major resources for dealing with disasters. There are now a number of studies of these phenomena. For example, Vieweg et al. (2010) studied the use of Twitter in two natural emergencies, the 2009 grass fires in Oklahoma and the 2009 flooding of the Red River. They found that the use of Twitter made major contributions to situational awareness in both cases. Palen et al. (2009) studied the role of CMC in the 2007 mass shooting at Virginia Tech. Mark, Al-Ani, and Semaan (2009) looked at the role of CMC in maintaining resilience in a war zone, particularly Iraq. Schafer, Ganoe, and Carroll (2007) looked at emergency management planning in a community, focusing on what kind of software architecture would support the kinds of needs faced there. These examples are just the tip of the iceberg of recent work in the area, all of which is showing that the wide variety of collaboration technologies now available can have a substantial impact on the handling of emergencies.

24.4 COORDINATION SUPPORT

24.4.1 Meeting Support

An early and popular topic in CSCW was the support of FTF meetings. A number of systems were developed and tested. While of late the focus has shifted to the support of geographically distributed meetings, the early work on meeting support led to some important and useful conclusions.

Some meeting-support software imposed structure on the process of the meeting, embodying various brainstorming and voting procedures. Group decision-support systems (GDSSs) arose from a number of business schools, focusing on large meetings of stakeholders' intent on going through a set series of decisions, such as prioritizing projects for future funding (Nunamaker et al. 1991). With the help of a facilitator and some technical support, the group was led through a series of stages: brainstorming without evaluating, evaluating alternatives from a variety of positions, prioritizing alternatives, and so on. These meetings were held in specialized rooms in which individual computers were embedded in the tables, networked to central services, and summary displays shown "center stage." A typical scenario involved individuals silently entering ideas into a central repository, and after a certain amount of time, they were shown ideas one at a time from others and asked to respond with a new idea triggered by that one. Later, these same ideas were presented to the individuals who were then asked to rank or rate them according to some fixed criterion, like cost. Aggregates of individuals' opinions were computed, discussed further and presented for vote. The system applied computational power (for voting and rating mechanisms), and networking control (for parallel input) to support typically weak aspects of meetings. These systems were intended to gather more ideas from participants, since one did not have to wait for another to stop speaking in order to get a turn. And, anonymous voting and rating was intended to insure equal participation, not dominated by those in power.

Evaluations of these GDSSs have been reviewed producing some generalizations about their value (McLeod 1992; Kraemer and Pinsoneault 1990; Hollingshead, McGrath, and O'Connor 1993). The systems indeed fulfill their intentions of producing more ideas in brainstorming and having more evaluative comments because of anonymity. Decisions are rated as higher in quality, but the meetings take longer and the participants are less satisfied than those in traditional meetings.

A second class of technologies to support real-time meetings is less structured, more similar to individual workstation support. In these systems, groups are allowed access to a single document or drawing, and can enter and edit into them simultaneously at will. Different systems enforce different "locking" mechanisms (e.g., paragraph or selection locking) so that one person does not enter while another deletes the same thing (Ellis, Gibbs, and Rein 1991). Some also allow parallel individual work, where participants view and edit different parts of the same document, but can also view and discuss the same part as well. This kind of unstructured shared editor has been shown to be very effective for certain kinds of free-flowing meetings, like design or requirements meetings (Olson et al. 1993). The rated quality of the meeting products (e.g., a requirements document or plan) was higher when using these technologies than with traditional whiteboard or paper-and-pencil support, but like working in GDSSs, people were slightly less satisfied. The lower satisfaction here and with GDSSs may reflect the newness of the technologies; people may not have yet learned how to persuade, negotiate, or influence each other in comfortable ways, to harness the powers inherent in the new technologies.

These new technologies did indeed change the way in which people worked. They talked less and wrote more, building on each other's ideas instead of generating far-reaching other ideas. The tool seemed to focus the groups on the core ideas, and keep them from going off on tangents. Many participants reported really liking doing work in the meetings rather than spending time only talking about the work (Olson et al. 1993).

A third class of meeting room support appears in electronic whiteboards. For example, the LiveBoard (Elrod et al. 1992), SoftBoard and SmartBoard are rear projection surfaces that allow pen input, much the way a whiteboard or flip chart does. People at Xerox PARC (Palo Alto Research Center Incorporated) and Boeing have evaluated the use of these boards in meetings in extended case studies. In both cases, the board was highly valued because of its computational power and the fact that all could see the changes as they were made. At both sites, successful use required a facilitator who was familiar with the applications running to support the meeting. At Xerox, suggestions made in the meeting about additional functionality were built into the system so that it eventually was finely tuned support for their particular needs (Moran et al. 1996). For example, they did a lot of list making of freehand text items. Eventually, the board software recognized the nature of a list and an outline, with simple gestures changing things sensibly. For example, if a freehand text item was moved higher in a list, the other items adjusted their positions to make room for it. The end product was not only a set of useful meeting tools, but also a toolkit to allow people to build new meeting widgets to support their particular tasks.

As technological developments have enabled the creation of large, affordable displays, research has picked up on the utility of large displays for small team collaboration. An obvious application of large displays is for complex, high-resolution data, such as maps, medical images, and a variety of complex scientific visualizations. There are of course interesting issues in dealing with such large displays, such as how to navigate them when they are extraordinarily rich in detailed information (Ball, North, and Bowman 2007), how to distribute control among users (e.g., single selection device vs. one per person; see Birnholtz et al. [2007]), and how to deal with sensitive or private information, as in the context of shift changes in a hospital (Wilson, Galliers, and Fone 2006). Robertson et al. (2005) have a good discussion of the many usability issues that arise with large displays. There is a large and growing literature on this topic.

Tabletop displays constitute another way of presenting lots of information for collaborators. This too is a rapidly growing area of research that we do not have space to cover in any detail. But some recent, representative studies that would help one get into the literature include Isenberg et al. (2010), Morris, Lombardo, and Wigdor (2010), Hartmann et al. (2009) and Tang et al. (2006).

Meetings are important, though often despised, organizational activities. Research of the kind just reviewed has shown quite clearly that well-designed tools can improve both work outcomes and participant satisfaction. However, meetings in organizations seldom use such tools. Inexpensive mobile computing and projection equipment combined with many commercial products mean that such tools are within reach of most organizations. But not having these elements readily available in an integrated way probably inhibits their widespread adoption.

While traditional meetings are often viewed as wasteful and frustrating, there can be huge benefits to working together in collocated environments. Kiesler and Cummings (2002) reviewed a number of the characteristics of physical collocation that can benefit performance. In a detailed study of one such situation, Teasley et al. (2002) found that "radical collocation," in which software development teams worked together in a dedicated project room for many weeks, dramatically improved their productivity. Reasons for this included the constant awareness of each other's work status, the associated ability to instantly work on an impasse as a group, and the availability of rich shared artifacts generated by the project.

24.4.2 Workflow

Workflow systems lend technology support to coordinated asynchronous (usually sequential) steps of activities among team members working on a particular task. For example, a workflow system might route a travel reimbursement voucher from the traveler to the approving party to the accounts payable to the bank. The electronic form would be edited and sent to the various parties, their individual to-do-lists updated as they received and/or completed the tasks, and permissions and approval granted automatically as appropriate (e.g., allowing small charges to an account if the charges had been budgeted previously or simply if there was enough money in the account). Not only is the transaction flow supported, but also records are often kept about who did what and when they did it. It is this later feature that has potentially large consequences for the people involved, discussed in the last paragraph in this section.

These workflow systems were often the result of work reengineering efforts, focusing on making the task take less time and to eliminate the work that could be automated. Not only do workflow systems therefore have a bad reputation in that they often are part of workforce reduction plans, but also for those left, their work is able to be monitored much more closely. The systems are often very rigid, requiring, for example, all of a form to be filled in before it can be handed off to the next in the chain. They often require a great deal of rework because of this inflexibility. It is because of the inflexibility and the potential monitoring that the systems fall into disuse (e.g., Abbott and Sarin 1994). However, Grinter (2000) examined several cases of successful deployment of workflow systems, and drew some helpful conclusions about what is required for these to work. Klein, Dellarocas, and Bernstein (2000) introduced a special double issue of *Computer Supported Cooperative Work* on adaptive workflow systems that deal with some of the exception-handling that can be such an important feature for success.

The fact that workflow can be monitored is a major source of user resistance. In Europe, such monitoring is illegal, and powerful groups of organized workers have made sure that such capabilities are not in workflow systems (Prinz and Kovenbach 1996). In the United States, it is not illegal, but many employees complain about its inappropriate use. For example, in one software engineering team where workflow had just been introduced to track bug reports and fixes, people in the chain were sloppy about noting who they had handed a piece of work off to. When it was discovered that the manager had been monitoring the timing of the handoffs to assign praise or blame, the team members were justifiably upset (Olson and Teasley 1996). In general, managerial monitoring is a feature that is not well received by people being monitored (Markus 1983). If such monitoring is mandated, workers' behavior will conform to the specifics of what is being monitored (e.g., time to pass an item off to the next in the chain) rather than perhaps to what the real goal is (e.g., quality as well as timely completion of the whole process).

24.4.3 Group Calendars

A number of organizations have now adopted online calendars, mainly in order to view people's schedules to arrange meetings. The calendars also allow a form of awareness, allowing people to see if a person who is not present is expected back soon. Individuals benefit only insofar as they offload scheduling meetings to others, like to an administrative assistant, who can write as well as read the calendar. And, in some systems the individual can schedule private time, blocking the time but not revealing to others his or her whereabouts. By this description, on-line calendaring is a classic case of what Grudin (1988) warned against, a misalignment of costs and benefits; the individual puts in the effort to record his/her appointments so that another, in this case a manager or coworker, can benefit from ease of scheduling. However, since the early introduction of electronic calendaring systems, many organizations have found successful adoption (Mosier and Tammaro 1997; Grudin and Palen 1995; Palen and Grudin 2002). Apparently such success requires a culture of sharing and accessibility, something that exists in some organizations and not others (Lange 1992; Ehrlich 1987). But today group calendars are a common piece of infrastructure in many settings (Miller 2009).

24.4.4 Awareness

In normal work, there are numerous occasions in which people find out casually whether others are in and, in some cases, what they are doing. A simple walk down the hall to a printer offers numerous glances into people's offices, noting where their coats are, whether others are talking, whether there is intense work at a computer, and so on. This kind of awareness is unavailable to workers who are remote. Some researchers have offered various technology solutions; some have allowed one to visually walk down the hall at the remote location, taking a 5-second glance into each passing office (Bellotti and Dourish 1997; Fish et al. 1993). Another similar system, called “Portholes,” provides periodic snapshots instead of full-motion video (Dourish and Bly 1992). Because of privacy implications, these systems have had mixed success. The places in which this succeeds are those in which the individuals seem to have a reciprocal need to be aware of each other’s presence, and a sense of cooperation and coordination. A contrasting case is the IM system in which the user has control as to what state they wish to advertise to their partners about their availability. The video systems are much more lightweight to the user but more intrusive; the IM ones give the user more control but require intention in action. Another approach investigated by Ackerman et al. (1997) looked at shared audio as an awareness tool, though this too has privacy implications.

As mentioned earlier, instant messaging systems provide an awareness capability. Most systems display a list of “buddies” and whether they are currently on-line or not. Nardi, Whittaker, and Bradner (2000) found that people liked this aspect of IM (see also Muller et al. [2003]; Isaacs et al. [2002]). And, since wireless has allowed constant connectivity of mobile devices like PDAs, this use of tracking others is likely to grow. But again, there are issues of monitoring for useful or insidious purposes, and the issues of trust and privacy loom large (see Godefroid et al. [2000]).

Another approach to signaling what one is doing occurs at the more micro level. And again, one captures what is easy to capture. When people are closely aligned in their work, there are applications that allow each to see exactly where in the shared document the other is working and what they are doing (Gutwin and Greenberg 1999). If one is working nearby the other, this signals perhaps a need to converse about the directions each is taking. Empirical evaluations have shown that such workspace awareness can facilitate task performance (Gutwin and Greenberg 1999).

Studies of attempts to carry out difficult intellectual work within geographically distributed organizations show that one of the larger costs of geographical distribution is the lack of awareness of what others are doing or whether they are even around (Herbsleb et al. 2000). Thus, useful and usable awareness tools that mesh well with trust and privacy concerns could be of enormous organizational importance. This is a rich research area for CSCW.

An important body of material on this topic appeared in a special issue of *Computer Supported Cooperative Work* in 2002. We do not have the space to engage the nine important articles published in this special issue, but anyone wanting to delve more deeply into this topic of necessity needs to digest this special issue. Schmidt’s (2002) article exploring the very concept of awareness itself certainly deserves attention.

A related problem that has recently received much attention is the matter of interruptions. Interruptions have the property that there is an asymmetry between the interrupter and the interrupted, in that the former seemingly has more control over the occasions of interruptions than the latter (Nardi and Whittaker 2002).* These issues become even more acute in distributed work, especially with weak awareness support. Given this, several investigators have explored with some success whether techniques drawn from statistical decision theory or machine learning could be used to figure out from sensor data whether a person is interruptible (e.g., Horvitz and Apacible 2003; Fogerty et al. 2005).

24.5 INFORMATION REPOSITORIES

24.5.1 Repositories of Shared Knowledge

In addition to sharing information generally on the web, in both public and intranet settings, there are applications that are explicitly built for knowledge sharing. The goal in most systems is to capture knowledge that can be reused by others, like instruction manuals, office procedures, training, and “boilerplates,” or templates of commonly constructed genres, like proposals or bids. Experience shows, however, that these systems are not easy wins. Again, similar to the case of the on-line calendaring systems described in Section 24.4.3, the person entering information into the system is not necessarily the one benefiting from it. In a large consulting firm, where consultants were quite competitive in their bid for advancement, there was indeed negative incentive for giving away one’s best secrets and insights (Orlikowski and Gash 1994).

Sometimes subtle design features are at work in the incentive structure. In another adoption of Lotus Notes, in this case to track open issues in software engineering, the engineers slowly lost interest in the system because they assumed that their manager was not paying attention to their contributions and use of the system. The system design, unfortunately, made the manager’s actual use invisible to the team. Had they known that he was reading daily what they wrote (though he never wrote anything himself), they would likely have continued to use the system (Olson and Teasley 1996). A simple design change that would make the manager’s reading activity visible to the team would likely have significantly altered their adoption.

The web of course provides marvelous infrastructure for the creation and sharing of information repositories. A variety of tools are appearing to support this. Of particular interest are open source tools that allow for a wider, more flexible infrastructure for supporting information sharing (see www.sakai.org). The major types of collaboratory (see Section 24.7.3) are those that provide shared data repositories for a community of scientists. The topic of “knowledge management” has received extensive treatment over the past decade or more and is far beyond the scope of what we can review here.†

* Though one interesting finding is that in the normal course of activity, many interruptions are self-administered (Mark, Gonzalez, and Harris 2005).

† A Google search in August 2010 under “knowledge management” yielded over 70 million hits!

24.5.2 Wikis

A wiki is a shared web space that can be edited by anyone who has access to it. They were first introduced by Ward Cunningham in 1995, but have recently become very popular. These can be used in a variety of ways, both for work and for fun. The most famous wiki is Wikipedia (www.wikipedia.org), an online encyclopedia where anyone can generate and edit content. It has grown to have millions of entries, and has versions in at least ten languages. A recent study carried out by *Nature* found that for science articles Wikipedia and the Encyclopedia Britannica were about equally accurate (Giles 2005). Bryant, Forte, and Bruckman (2005) studied the contributors to Wikipedia, and suggested that a new publishing paradigm was emerging. Viegas, Wattenberg, and Dave (2004) developed imaginative visualizations of Wikipedia authoring and editing behavior over time. While an extensive literature on Wikipedia has developed in recent years, an interesting paper by Kittur and Kraut (2010) studies nearly 7000 other wikis, noting both similarities and differences with the findings that have emerged from studies of Wikipedia. For instance, coordination mechanisms across a wide range of wikis tended to be similar to Wikipedia. But a wide range of policies, procedures, and other mechanisms for managing a wiki appeared in the larger sample.

24.5.3 Capture and Replay

Tools that support collaborative activity can create traces of that activity that later can be replayed and reflected upon. The UARC explored the replay of earlier scientific campaign sessions (Olson et al. 2001), so that scientists could reflect upon their reactions to real-time observations of earlier phenomena. Using a video cassette recording metaphor, they could pause where needed, and fast forward past uninteresting parts. This reflective activity could also engage new players who had not been part of the original session. Abowd (1999) has explored such capture phenomena in an educational experiment called Classroom 2000. Initial experiments focused on reusing educational sessions during the term in college courses. Lipford and Abowd (2008) report on the long-term deployment of such a system, noting a number of challenges in making such systems effective. We do net yet fully understand the impact of such promising ideas.

24.6 SOCIAL COMPUTING

24.6.1 Social Filtering, Recommender Systems

We often find the information we want by contacting others. Social networks embody rich repositories of useful information on a variety of topics. A number of investigators have looked at whether the process of finding information through others can be automated. The kinds of recommender systems that we find on websites like Amazon.com are examples of the result of such research. The basic principle of such systems is that an individual will tend to like or prefer the kinds of things (e.g., movies and books) that someone who is similar to him/her likes. They find similar people by matching their previous choices. Such systems use a variety of algorithms to match preferences with those of others, and then recommend new items. Resnick and Varian (1997) edited a special issue of the *Communication of the ACM* on recommender systems that included a representative set of examples. Herlocker, Konstan, and Riedl (2000) used empirical methods to explicate the factors that led users to accept the advice of recommender systems. In short, providing access to explanations for why items were recommended seems to be the key. Cosley et al. (2005) studied factors that influence people to contribute data to recommender systems. Recommender systems are emerging as a key element of e-commerce (Schafer, Konstan, and Riedl 2001). Accepting the output of recommender systems is an example of how people come to trust technical systems. This is a complex topic, and relates to issues like security that we briefly described in Section 24.3.

24.6.2 Trust of People via the Technology

It has been said that “trust needs touch,” and indeed in survey studies, coworkers report that they trust those who are collocated more than those who are remote (Rocco et al. 2000). Interestingly, those who spend the most time on the phone chatting about non-work-related topics with their remote coworkers show higher trust than those they communicate with using only fax and e-mail. But lab studies show that telephone interaction is not as good as FTF. People using just the telephone behave in more self-serving, less-trusting ways than they do when they meet face to face (Drolet and Morris 2000).

What can be done to counteract the mistrust that comes from the impoverished media? Rocco (1998) had people meet and do a team-building exercise the day before they engaged in the social dilemma game with only e-mail to communicate with. These people, happily, showed as much cooperation and trust as those who discussed things face to face during the game. This is important. It suggests that if remote teams can do some FTF teambuilding before launching on their project, they will act in a trusting/trustworthy manner.

Since it is not always possible to have everyone on a project meet face to face before they launch into the work, what else will work? Researchers have tried some options, but with mixed success. Zheng et al. (2001) found that using chat for socializing and sharing pictures of each other also led to trustful relations. Merely sharing a resume did not. When the text is translated into voice, it has no effect on trust, and when it is translated into voice and presented in a moving human-like face, it is even worse than text-chat. (Jensen et al. 2000; Kiesler, Sproull, and Waters 1996). However, Bos et al. (2001) found that interactions over video and audio led to trust, albeit of a seemingly more fragile form.

If we can find a way to establish trust without expensive travel, we are likely to see important productivity gains. Clearly the story is not over. However, we must not be too optimistic. In other tasks, video does not produce “being there.” There is an overhead to the conversation through video; it requires more effort than working face to face

(Olson, Olson, and Meader 1995). And, today's videos over the Internet are both delayed and choppy, producing cues that people often associate with lying. One does not trust someone who appears to be lying. Trust is a delicate emotion; today's video might not just do it in a robust enough fashion, though the Nguyen and Canny (2007) study mentioned in Section 24.3.3 presented some encouraging results.

24.7 INTEGRATED SYSTEMS

24.7.1 Media Spaces

As an extension of video conferencing and awareness systems, some people have experimented with open, continuous audio and video connections between remote locations. In a number of cases, these experiments have been called "Media Spaces," and these were very popular experiments in industry in the late 1980s and early 1990s. For example, at Xerox, two labs were linked with an open video link between two commons areas (Olson and Bly 1991), the two locations being Palo Alto, California, and Portland, Oregon. Evaluation of these experiments showed that maintaining organizational cohesiveness at a distance was much more difficult than when members are collocated (Finn, Sellen, and Wilbur 1997). However, some connectedness was maintained. Where many of these early systems were plagued with technical difficulties, human factors limitations, or very large communication costs, in today's situation it might actually be possible to overcome these difficulties, making media a possibility for connecting global organizations. A new round of experimental deployments with new tools is needed.

24.7.2 Collaborative Virtual Environments

Collaborative virtual environments are 3D embodiments of multiuser domains (MUDs). The space in which people interact is an analog of physical space, with dimensions, directions, rooms, and objects of various kinds. People are represented as avatars—simplified, geometric, digital representations of people, who move about in the 3D space (Singhal and Zyda 1999). Similar to MUDs, the users in a meeting situation might interact over some object that is digitally represented, like a mock up of a real thing (e.g., an automobile engine, an airplane hinge, a piece of industrial equipment) or with visualizations of abstract data (e.g., a 3D visualization of atmospheric data). In these spaces, one can have a sense as to where others are and what they are doing, similar to the simplified awareness systems described earlier. In use, it is difficult to establish mutual awareness or orientation in such spaces (Hindmarsh et al. 1998; Park, Kapoor, and Leigh 2000; Yang and Olson 2002). There have even been some attempts to merge collaborative virtual environments with real ones, though with limited success so far (Benford et al. 1998).

The emergence of multiplayer games with rich virtual environments has literally exploded in the past decade. There is a growing literature on the characteristics of play and collaboration in these games. Ducheneaut and Moore (2005) described the learning of social skills in such games. Brown and Thomas (2006) speculated that achieving mastery in such collaborative games might become an important entry on a resume. Bainbridge (2007) discussed the scientific research potential of such worlds. Nardi (2010) summarized her extensive experience in playing World of Warcraft. This is just the briefest sample of what is available.

Another recent development is the emergence of virtual environments like Second Life. These are not game environments, but a platform in which a wide range of social phenomena are supported in a virtual world. Participants have an avatar, whose visual appearance and clothing can be designed. People in Second Life engage in commerce, buying and selling real estate, making things like clothing or furniture, and a variety of other imports from real life (or RL as it is known in Second Life). Many colleges and universities have a presence in Second Life, and have experimented with offering classes and discussion forums. A number of corporations have a presence in Second Life, and have engaged in creative activities such as prototyping future places (e.g., hotel designs) or software. The first author of this chapter recently participated in a usability evaluation of software developed by IBM Research.

24.7.3 Collaboratories

A collaboratory is a laboratory without walls (Finholt and Olson 1997). From a National Research Council report, a collaboratory is supposed to allow "the nation's researchers [to] perform their research without regard to geographical location—interacting with colleagues, accessing instrumentation, sharing data and computational resources [and] accessing information in digital libraries" (National Research Council 1993, p. 7). Starting in the early 1990s, these capabilities have been configured into support packages for a number of specific sciences (see Finholt [2002]). The Science of Collaboratories project (www.science of collaboratories.org) has identified more than 200 existing collaboratories and has drawn lessons about why some succeed and others do not (Olson, Zimmerman, and Bos 2008).

A number of companies have also experimented with similar concepts, calling them "virtual collocation." The goal there is to support geographically dispersed teams as they carry out product design, software engineering, financial reporting, and almost any business function. In these cases, suites of off-the-shelf groupware tools have been particularly important and have been used to support round-the-clock software development among overlapping teams of engineers in time zones around the world. (Carmel 1999). There have been a number of such efforts, and it is still unclear as to their success or what features make their success more likely, though Olson et al. (2008) have summarized a rich variety of possible factors.

24.8 CONCLUSIONS

Many of the functions we have described in this chapter are becoming ordinary elements of infrastructure in networked computing systems. Prognosticators looking at the emergence of collaborative technologies and the convergence of computing and communication media have forecast that distance will diminish as a factor in human interactions (e.g., Cairncross 1997). However, to paraphrase Mark Twain, the reports of distance's death are greatly exaggerated. Even with all our emerging information and communications technologies, distance and its associated attributes of culture, time zones, geography, and language will continue to affect how humans interact with each other. Emerging distance technologies will allow greater flexibility for those whose work must be done at a distance, but we believe (see Olson and Olson [2000]) that distance will continue to be a factor in understanding these work relationships.

ACKNOWLEDGMENTS

Preparation of this chapter was facilitated by several grants from the National Science Foundation (research grants IIS-9320543, IIS-9977923, ATM-9873025, IIS-0085951 and cooperative agreement IRI-9216848) and a grant from the Army Research Institute (W91WAW-07-C-0060). We are also grateful to several anonymous reviewers for helpful comments on earlier drafts.

REFERENCES

Abbate, J. 1999. *Inventing the Internet*. Cambridge, MA: MIT Press.

Abbott, K. R., and S. K. Sarin. 1994. Experiences with workflow management: Issues for the next generation. In *Proceedings of the Conference on Computer Supported Cooperative Work*, 113–20. Chapel Hill, NC: ACM Press.

Abowd, G. D. 1999. Classroom 2000: An experiment with the instrumentation of a living educational environment. *IBM Syst J* 38:508–30.

Ackerman, M. S., B. Starr, D. Hindus, and S. D. Mainwaring. 1997. Hanging on the 'wire': A field study of an audio-only media space. *ACM Trans Comput Hum Interact* 4(1):39–66.

Adamic, L. A., and N. Glance. 2005. The political blogosphere and the 2004 U.S. election: Divided they blog. Presented at LinkKDD-2005, Chicago, IL. www.blogpulse.com/papers/2005/ (accessed March 13, 2007).

Anderson, R. H., T. K. Bikson, S. A. Law, and B. M. Mitchell. 1995. *Universal Access to E-Mail: Feasibility and Societal Implications*. Santa Monica, CA: Rand.

Argyle, M., and M. Cook. 1976. *Gaze and Mutual Gaze*. New York, NY: Cambridge University Press.

Arrow, H., J. L. Berdahl, K. S. Bouas, K. M. Craig, A. Cummings, L. Lebei, J. E. McGrath, K. M. O'Connor, J. A. Rhoades, and A. Schlosser. 1996. Time, technology, and groups: An integration. *Comput Support Coop Work* 4:253–61.

Atkins, D. E., K. K. Droegemeier, S. I. Feldman, H. Garcia-Molina, M. L. Klein, D. G. Messerschmitt, P. Messina, J. P. Ostriker, and M. H. Wright. 2003. Revolutioning science and engineering through cyberinfrastructure. Report of the National Science Foundation Blue-Ribbon Advisory Panel on Cyberinfrastructure. National Science Foundation. Arlington, VA.

Baecker, R. M. 1993. *Readings in Groupware and Computer-Supported Cooperative Work*. San Mateo, CA: Morgan Kaufmann Publishers.

Bainbridge, W. S. 2007. The scientific research potential of virtual worlds. *Science* 317:472–6.

Ball, R., C. North, and D. A. Bowman. 2007. Move to improve: Promoting physical navigation to increase user performance with large displays. In *Proceedings of CHI 2007*, 191–200. New York: ACM.

Bargh, J. A., and K. Y. A. McKenna. 2004. The Internet and social life. *Annu Rev Psychol* 55:573–90.

Bell, A. 2009. *Exploring Web 2.0: Second Generation Interactive Tools—Blogs, Podcasts, Wikis, Networking, Virtual Worlds, and More*. Georgetown, TX: Katy Crossings Press.

Bellotti, V., and P. Dourish. 1997. Rant and RAVE: Experimental and experiential accounts of a media space. In *Video-Mediated Communication*, ed. K. E. Finn, A. J. Sellen, and S. B. Wilbur, 245–72. Mahwah, NJ: Lawrence Erlbaum Associates.

Bellotti, V., N. Ducheneaut, M. Howard, I. Smith, and R. E. Grinter. 2005. Quality versus quantity: E-mail centric task management and its relation with overload. *Hum Comput Interact* 20:89–138.

Benford, S., C. Greenhalgh, G. Reynard, C. Brown, and B. Koleva. 1998. Understanding and constructing shared spaces with mixed-reality boundaries. *ACM Trans Comput Hum Interact* 5:185–223.

Benkler, Y. 2006. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. New Have, CT: Yale University Press.

Berners-Lee, T. 1999. *Weaving the Web*. New York: Harper Collins.

Birnholtz, J. P., T. Grossman, C. Mak, and R. Balakrishnan. 2007. An exploratory study of input configuration and group process in a negotiation task using a large display. In *Proceedings of CHI 2007*, 91–100. New York: ACM.

Bos, N., D. Gergle, J. S. Olson, and G. M. Olson. 2001. Being there vs. seeing there: Trust via video. Short paper presented at the Conference on Human Factors in Computing Systems: CHI-2001. New York: ACM Press.

Brown, J. S., and D. Thomas. 2006. You play World of Warcraft? You're hired. *Wired Magazine*, 14(4).

Bryant, S. L., A. Forte, and A. Bruckman. 2005. Becoming Wikipedian: Transformation of participation in a collaborative online encyclopedia. In *Proceedings of GROUP 2005*, 1–10. New York: ACM.

Bush, V. 1945. As we may think. *Atl Mon* 176(1):101–8.

Cadiz, J., A. Balachandran, E. Sanocki, A. Gupta, J. Grudin, and G. Jancke. 2000. Video conferencing as a technology to support group work: A review of its failure. Paper presented at the CSCW '88, New York: ACM Press.

Cairncross, F. 1997 *The Death of Distance: How the Communications Revolution Will Change Our Lives.* Boston, MA: Harvard Business School Press.

Camp, L. J. 2000. *Trust and Risk in Internet Commerce*. Cambridge, MA: MIT Press.

Campasoto, O., and K. Nilson. 2011. *Web 2.0 Fundamentals for Developers: With AJAX, Development Tools, and Mobile Platforms*. Sudbury, MA: Jones & Bartlett Publishers.

Carley, K., and K. Wendt. 1991. Electronic mail and scientific communication: A study of the Soar extended research group. *Knowl Creation Diffus Util* 12:406–40.

Carmel, E. 1999. *Global Software Teams*. Upper Saddle River, NJ: Prentice-Hall.

Comer, D. E. 2008. *Computer Networks and Internets*. Englewood Cliffs, NJ: Prentice-Hall.

Connaughton, S. L., and M. Shuffler. 2007. Multinational and multicultural distributed teams: A review and future agenda. *Small Group Res* 38:387–412.

Cosley, D., D. Frankowski, S. Kiesler, L. Terveen, and J. Riedl. 2005. How oversight improves member-maintained communities. In *Proceedings of CHI 2005*, 11–20. New York: ACM.

Crabtree, A., T. Rodden, and S. Benford. 2005. Moving with the times: IT research and the boundaries of CSCW. *Comput Support Coop Work* 14:217–51.

Daly-Jones, O., A. Monk, and L. Watts. 1998. Some advantages of video conferencing over high-quality audio conferencing: Fluency and awareness of attentional focus. *Int J Hum Comput Stud* 49(1):21–58.

DeSanctis, G., B. M. Jackson, M. S. Poole, and G. W. Dickson. 1996. Infrastructure for telework: Electronic communication at Texaco. In *Proceedings of SIGCPR/SIGMIS '96*, 94–102. New York: ACM.

Diamant, E. I., S. R. Fussell, and F.-L. Lo. 2009. Collaborating across cultural and technological boundaries: Team culture and information use in a map navigation task. In *Proceedings of the ACM International Workshop on Intercultural Collaboration (IWIC 2009)*, 175–84. New York: ACM.

Dourish, P., and S. Bly. 1992. Portholes: Supporting awareness in a distributed work group. In *Proceedings of CHI '92*, 541–7. New York: ACM Press.

Drolet, A. L., and M. W. Morris. 2000. Rapport in conflict resolution: Accounting for how nonverbal exchange fosters coordination on mutually beneficial settlements to mixed motive conflicts. *J Exp Soc Psychol* 36(1):26–50.

Ducheneaut, N., and R. J. Moore. 2005. More than just 'XP': Learning social skills in massively multiplayer online games. *Interact Technol Smart Educ* 2:89–100.

Egido, C. 1988. Video conferencing as a technology to support group work: A review of its failure. Paper presented at the CSCW '88, New York: ACM Press.

Ehrlich, S. F. 1987. Strategies for encouraging successful adoption of office communication systems. *ACM Trans Office Inf Syst* 5:340–57.

Ellis, C. A., S. J. Gibbs, and G. L. Rein. 1991. Groupware: Some issues and experiences. *CACM* 34(1):38–58.

Ellis, C. A., and G. Nutt. 1980. Office information systems and computer science. *Comput Surv* 12(1):27–60.

Elrod, S., R. Bruce, R. Gold, D. Goldberg, F. Halasz, W. Janssen, D. Lee et al. 1992. LiveBoard: A large interactive display supporting group meetings, presentations, and remote collaboration. In *Proceedings of CHI '92*, 599–607. Monterey, CA: ACM Press.

Engelbart, D., and W. English. 1968. A research center for augmenting human intellect. *Proc FJCC* 33:395–410.

Federal Trade Commission. 2005. E-mail address harvesting and the effectiveness of anti-spam filters. Report by Federal Trade Commission, Division of Marketing Practices. Washington, DC.

Finholt, T. A. 2002. Collaboratories. In *Annual Review of Information Science and Technology*, ed. B. Cronin, 74–107. Washington, DC: American Society for Information Science.

Finholt, T. A., and G. M. Olson. 1997. From laboratories to collaboratories: A new organizational form for scientific collaboration. *Psychol Sci* 8:28–36.

Finholt, T., L. Sproull, and S. Kiesler. 1990. Communication and performance in ad hoc task groups. In *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work,* ed. J. Galegher, R. Kraut, and C. Egido, 291–325. Hillsdale, NJ: Lawrence Erlbaum Associates.

Finn, K., A. Sellen, and S. Wilbur, eds. 1997. *Video-Mediated Communication*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Fish, R. S., R. E. Kraut, R. W. Root, and R. E. Rice. 1993. Video as a technology for informal communication. *Commun ACM* 36(1):48–61.

Fogerty, J., S. E. Hudson, C. G. Atkeson, D. Avrahami, J. Forlizzi, S. Kiesler, J. C. Lee, and J. Yang. 2005. Predicting human interruptibility with sensors. *ACM Trans Comput Hum Interact* 12:119–46.

Foster, I., and C. Kesselman. 2004. *The Grid: Blueprint for a New Computing Infrastructure*. 2nd ed. San Francisco: Morgan Kaufmann.

Fussell, S. R., R. E. Kraut, and J. Siegel. 2000. Coordination of communication: Effects of shared visual context on collaborative work. Paper presented at the CSCW 2000, New York: ACM Press.

Gale, C., and A. Monk. 2000. Where am I looking? The accuracy of video-mediated gaze awareness. Percept Psychophys 62:586–95.

Giles, J. 2005. Internet encyclopaedias go head to head. *Nature* 438:900–1.

Godefroid, P., J. D. Herbsleb, L. J. Jagadeesan, and D. Li. 2000. Ensuring privacy in presence awareness systems: An automated verification approach. In *Proceedings of CSCW 2000*, 59–68. New York: ACM.

Grayson, D. M., and A. Monk. 2003. Are you looking at me? Eye contact and desktop video conferencing. *ACM Trans Comput Hum Interact* 10(3):221–43.

Greif, I., ed. 1988. *Computer-Supported Cooperative Work: A Book of Readings*. San Mateo, CA: Morgan Kaufmann.

Grinter, R. E. 2000. Workflow systems: Occasions for success and failure. *Comput Support Coop Work* 9:189–214.

Grudin, J. 1988. Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces. In *Proceedings of the Conference on Computer Supported Cooperative Work*, 85–93. Portland, OR: ACM Press.

Grudin, J. 1994. Groupware and social dynamics: Eight challenges for developers. *Commun ACM* 37(1):92–105.

Grudin, J., and L. Palen. 1995. Why groupware succeeds: Discretion or mandate? In *Proceedings of the European Computer Supported Cooperative Work*, 263–78. Stockholm, Sweden: Springer.

Grudin, J., and S. Poltrock. 2011. Taxonomy and theory in computer supported cooperative work. In *Oxford Handbook of Organizational Psychology*, ed. S. W. J. Kozlowski. New York: Oxford University Press.

Gunaratne, J. A., and A. J. B. Brush. 2010. Newport: Enabling sharing during mobile calls. In *Proceedings of CHI 2010*, 343–52. New York: ACM.

Gutwin, C., and S. Greenberg. 1999. The effects of workspace awareness support on the usability of real-time distributed groupware. *ACM Trans Comput Hum Interact* 6:243–81.

Hartmann, B., M. R. Morris, H. Benko, and A. D. Wilson. 2009. Augmenting interactive tables with mice & keyboards. In *Proceedings of UIST 2009*, 149–52. New York: ACM.

Herbsleb, J. D., A. Mockus, T. A. Finholt, and R. E. Grinter. 2000. Distance, dependencies, and delay in a global collaboration. In *Proceedings of CSCW 2000*, 319–28. New York: ACM.

Herlocker, J. L., J. A. Konstan, and J. Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of CSCW 2000*, 241–208250. New York: ACM.

Hindmarsh, J., M. Fraser, C. Heath, S. Benford, and C. Greenhalgh. 1998. Fragmented Interaction: Establishing Mutual Orientation in Virtual Environments. In *Proceedings of Conference on Computer-Supported Cooperative Work*, 217–26. Portland, OR: ACM Press.

Hollingshead, A. B., J. E. McGrath, and K. M. O'Connor. 1993. Group performance and communication technology: A longitudinal study of computer-mediated versus face-to-face work. *Small Group Res* 24:307–33.

Horvitz, E., and J. Apacible. 2003. Learning and reasoning about interruption. In *Proceedings of the International Conference on Multimodal Interfaces*, 20–7. New York: ACM.

Huang, W., J. S. Olson, and G. M. Olson. 2002. Camera angle affects dominance in video-mediated communication. Paper presented at the CHI 2002, New York: ACM Press.

Huberman, B., D. M. Romero, and F. Wu. 2009. Social networks that matter: Twitter under the microscope. *First Monday*, 14(1).

Isaacs, E., A. Walendowski, S. Whittaker, D. J. Schiano, and C. Kamm. 2002. The character, functions, and styles of instant messaging in the workplace. Paper presented at the CSCW 2002, New York: ACM Press.

Isenberg, P., D. Fisher, M. R. Morris, K. Inkpen, and M. Czerwinski. 2010. An exploratory study of co-located collaborative visual analytics around a tabletop display. In *Proceedings of Visual Analytics Science and Technology*. Los Alamitos, CA: IEEE Computer Society.

Jensen, C., S. D. Farnham, S. M. Drucker, and P. Kollock. 2000. The effect of communication modality on cooperation in on-line environments. In *Proceedings of CHI '2000*, 470–7. New York: ACM Press.

Jones, M., and G. Marsden. 2006. *Mobile Interaction Design*. New York: Wiley.

Karsenty, L. 1999. Cooperative work and shared visual context: An empirical study of comprehension problems in side-by-side and remote help dialogues. *Hum Comput Interact* 14:283–315.

Kellogg, W. A., et al. 2006. Leveraging digital backchannels to enhance user experience in electronically mediated communication. In *Proceedings of CSCW 2006*, 451–4. New York: ACM.

Kendon, A. 1967. Some functions of gaze direction in social interaction. *Acta Psychol* 26:22–63.

Kiesler, S., and J. N. Cummings. 2002. What do we know about proximity and distance in work groups? A legacy of research. In *Distributed Work*, ed. P. J. Hinds and S. Kiesler, 57–80. Cambridge, MA: MIT Press.

Kiesler, S., L. Sproull, and K. Waters. 1996. Prisoner's dilemma experiment on cooperation with people and human-like computers. *J Pers Soc Psychol* 70(1):47–65.

Kittur, A., and R. E. Kraut. 2010. Beyond Wikipedia: Coordination and conflict in online production groups. In *Proceesings of CSCW 2010*, 215–24. New York: ACM.

Klein, M., C. Dellarocas, and A. Bernstein. 2000. Introduction to the special issue on adaptive workflow systems. *Comput Support Coop Work* 9:265–7.

Kraemer, K. L., and A. Pinsonneault. 1990. Technology and groups: Assessments of empirical research. In *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, ed. J. Galegher, R Kraut, and C. Egido, 373–405. Hillsdale, NJ: Lawrence Erlbaum Associates.

Krauss, R. M., and P. D. Bricker. 1966. Effects of transmission delay and access delay on the efficiency of verbal communication. *J Acoust Soc* 41:286–92.

Kraut, R. E., S. R. Fussell, and J. Siegel. 2003. Visual information as a conversational resource in collaborative physical tasks. *Hum Comput Interact* 18(1–2):13–39.

Kraut, R. E., D. Gergle, and S. R. Fussell. 2002. The use of visual information in shared visual spaces: Informing the development of virtual co-presence. Paper presented at the CSCW 2002, New York: ACM Press.

Kraut, R., S. Kiesler, B. Boneva, J. Cummings, V. Helgeson, and A. Crawford. 2002. Internet paradox revisited. *J Soc Issues* 58(1):49–74.

Kraut, R., M. Patterson, V. Lundmark, S. Kiesler, T. Mukopadhyay, and W. Scherlis. 1998. Internet paradox: A social technology that reduces social involvement and psychological well-being. *Am Psychol* 53:1017–31.

Kraut, R. E., S. Sunder, R. Telang, and J. Morris. 2005. Pricing electronic mail to solve the problem of spam. *Hum Comput Interact* 20:195–223.

Kurose, J. F., and K. W. Ross. 2009. *Computer Networking: A Top Down Approach*. 5th Ed. Reading, MA: Addison-Wesley.

Lange, B. M. 1992. Electronic group calendaring: Experiences and expectations. In *Groupware*, ed. D. Coleman, 428–32. San Mateo, CA: Morgan Kaufmann.

Ling, R., and J. Donner. 2009. *Mobile Phones and Mobile Communications*. Malden, MA: Polity Press.

Lipford, H. R., and G. D. Abowd. 2008. Reviewing meetings in TeamSpace. *Hum Comput Interact* 23:406–32.

Longstaff, T. A., J. T. Ellis, S. V. Hernan, H. F. Lipson, R. D. Mc Millan, L. H. Pesanti, and D. Simmel. 1997. Security on the Internet. In *The Froehlich/Kent Encyclopedia of Telecommunications*, Vol. 15, 231–55. New York: Marcel Dekker.

Luff, P., C. Heath, H. Kuzuoka, J. Hindmarsh, and S. Oyama. 2003. Fractured ecologies: Creating environments for collaboration. *Hum Comput Interact* 18(1–2):51–84.

Mackay, W. E. 1989. Diversity in the use of electronic mail: A preliminary inquiry. *ACM Trans Office Inf Syst* 6:380–97.

Malone, T. W., K. R. Grant, K. Y. Lai, R. Rao, and D. A. Rosenblitt. 1989. The information lens: An intelligent system for information sharing and coordination. In *Technological Support for Work Group Collaboration*, ed. M. H. Olson, 65–88. Hillsdale, NJ: Lawrence Erlbaum Associates.

Marca, D., and G. Bock. 1992. *Groupware: Software for Computer-Supported Cooperative Work*. Los Alamitos, CA: IEEE Computer Society Press.

Mark, G., B. Al-Ani, and B. Semaan. 2009. Resilience through technology adoption: Merging the old and the new in Iraq. In *Proceedings of CHI 2009*, 689–98. New York: ACM.

Mark, G., V. M. Gonzalez, and J. Harris. 2005. No task left behind? Examining the nature of fragmented work. In *Proceedings of CHI 2005*, 321–30. New York: ACM.

Markus, M. L. 1983. *Systems in Organization: Bugs and Features*. San Jose, CA: Pitman.

Markus, M. L., and T. Connolly. 1990. Why CSCW applications fail: Problems in the adoption of interdependent work tools. In *Proceedings of the Conference on Computer Supported Cooperative Work*, 371–80. Los Angeles, CA: ACM Press.

McDaniel, S. E., G. M. Olson, and J. S. Magee. 1996. Identifying and analyzing multiple threads in computer-mediated and face-to-face conversations. In *Proceeding of the ACM Conference on Computer Supported Cooperative Work*, 39–47. Cambridge, MA: ACM Press.

McLeod, P. L. 1992. An assessment of the experimental literature on electronic support of group work: Results of a meta-analysis. *Hum Comput Interact* 7:257–80.

Miller, M. 2009. *Cloud Computing: Web-Based Applications that Change the Way You Work and Collaborate Online*. Indianapolis, IN: Cue Publishing.

Monk, A., and C. Gale. 2002. A look is worth a thousand words: Full gaze awareness in video-mediated communication. *Discourse Process* 33(3):257–78.

Moran, T. P., P. Chiu, S. Harrison, G. Kurtenbach, S. Minneman, and W. van Melle. 1996. Evolutionary engagement in an ongoing collaborative work process: A case study. In *Proceeding of the ACM Conference on Computer Supported Cooperative Work*, 150–9. Cambridge, MA: ACM Press.

Morris, M. R., J. Lombardo, and D. Wigdor. 2010. WeSearch: Supporting collaborative search and sensemaking on a tabletop display. In *Proceedings of CSCW 2010*, 401–10. New York: ACM.

Mosier, J. N., and S. G. Tammaro. 1997. When are group scheduling tools useful? *Comput Support Coop Work* 6:53–70.

Muller, M. J., M. E. Raven, S. Kogan, D. R. Millen, and K. Carey. 2003. Introducing chat into business organizations: Toward an instant messaging maturity model. Paper presented at the GROUP '03, New York: ACM Press.

Nardi, B. A. 2010. *My Life as a Night elf Priest: An Anthropological Account of World of Warcraft*. Ann Arbor, MI: University of Michigan Press.

Nardi, B. A., D. J. Schiano, M. Gumbrecht, and L. Swartz. 2004. Why we blog? *Commun ACM* 47(12):41–6.

Nardi, B. A., H. Schwarz, A. Kuchinsky, R. Leichner, S. Whittaker, and R. Sclabassi. 1993. Turning away from talking heads: the use of video-as-data in neurosurgery. Paper presented at the CHI '93, New York: ACM Press.

Nardi, B. A., and S. Whittaker. 2002. The place of face-to-face communication in distributed work. In *Distributed Work*, ed. P. Hinds and S. Kiesler, 83–110. Cambridge, MA: MIT Press.

Nardi, B. A., S. Whittaker, and E. Bradner. 2000. Interaction and outeraction: Instant messaging in action. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, 79–88. Philadelphia, PA: ACM Press.

National Research Council. 1993. *National Collaboratories: Applying Information Technology for Scientific Research*. Washington, DC: National Academy Press.

Nguyen, D., and J. Canny. 2007. MultiView: Improving trust in group video conferencing through spatial faithfulness. In *Proceedings of CSCW 2007*, 1465–74. New York: ACM.

Nohria, N., and R. G. Eccles, eds. 1992. *Networks and Organizations: Structure, Form, and Action*. Boston, MA: Harvard Business School Press.

Nunamaker, J. F., A. R. Dennis, J. S. Valacich, D. R. Vogel, and J. F. George. 1991. Electronic meeting systems to support group work. *Commun ACM* 34(7):40–61.

O'Conaill, B., S. Whittaker, and S. Wilbur. 1993. Conversations over videoconferences: An evaluation of the spoken aspects of video mediated communication. *Hum Comput Interact* 8:389–428.

O'Hara-Devereaux, M., and R. Johansen. 1994. *Global Work: Bridging Distance, Culture & Time*. San Francisco, CA: Jossey-Bass.

Okada, K., F. Maeda, Y. Ichicawaa, and Y. Matsushita. 1994. Multiparty videoconferencing at virtual social distance: MAJIC design. Paper presented at the CSCW '94, New York: ACM Press.

Olson, G. M., D. Atkins, R. Clauer, T. Weymouth, A. Prakash, T. Finholt, F. Jahanian, and C. Rasmussen. 2001. Technology to support distributed team science: The first phase of the Upper Atmospheric Research Collaboratory (UARC) In *Coordination Theory and Collaboration Technology,* ed. G. M. Olson, T. Malone, and J. Smith, 761–83. Hillsdale, NJ: Lawrence Erlbaum Associates.

Olson, M. H., and S. A. Bly. 1991. The Portland experience: A report on a distributed research group. *Int J Man Mach Stud* 34:211–28.

Olson, J. S., E. C. Hofer, N. Bos, A. Zimmerman, G. M. Olson, D. Cooney, and I. Faniel. 2008. A theory of remote scientific collaboration (TORSC). In *Scientific Collaboration on the Internet*, ed. G. M. Olson, A. Zimmerman, & N. Bos. Cambridge, MA: MIT Press.

Olson, G. M., and J. S. Olson. 2000. Distance matters. *Hum Comput Interact* 15:139–79.

Olson, J. S., G. M. Olson, and D. K. Meader. 1995. What mix of video and audio is useful for remote real-time work? In *Proceedings of CHI '95*, 362–8. Denver, CO: ACM Press.

Olson, J. S., G. M. Olson, M. Storrøsten, and M. Carter. 1993. Group work close up: A comparison of the group design process with and without a simple group editor. *ACM Trans Inform Syst* 11:321–48.

Olson, J. S., and S. Teasley. 1996. Groupware in the wild: Lessons learned from a year of virtual collocation. In *Proceeding of the ACM Conference on Computer Supported Cooperative Work*, 419–27. Cambridge, MA: ACM Press.

Olson, G. M., A. Zimmerman, and N. Bos. 2008. *Scientific Collaboration on the Internet*. Cambridge, MA: MIT Press.

Orlikowski, W. J., and D. C. Gash. 1994. Technological frames: Making sense of information technology in organizations. *ACM Trans Inf Syst* 12:174–207.

Palen, L., and J. Grudin. 2002. Discretionary adoption of group support software: Lessons from calendar applications. In *Implementing Collaboration Technologies in Industry*, ed. B. E. Munkvold, 159–80. London: Springer-Verlag.

Palen, L., S. Vieweg, S. Liu, and A. Hughes. 2009. Crisis in a networked world: Features of computer-mediated communication the April 16, 2007, Virginia Tech event. *Social Science Computing Review* 27(4):467–80.

Park, K. S., A. Kapoor, and J. Leigh. 2000. Lessons learned from employing multiple perspective in a collaborative virtual environment for visualizing scientific data. In *Proceedings of ACM CVE '2000 Conference on Collaborative Virtual Environments*, 73–82. San Francisco, CA: ACM Press.

Prinz, W., and S. Kolvenbach. 1996. Support for workflows in a ministerial environment. In *Proceedings of the Conference on Computer Supported Cooperative Work*, 199–208. Cambridge, MA: ACM Press.

Putnam, R. D. 2000. *Bowling Alone: The Collapse and Revival of American Community*. New York: Simon & Schuster.

Resnick, P. 2002. Beyond bowling together: Sociotechnical capital. In *Human-Computer Interaction in the New Millennium*, ed. J. M. Carroll, 647–72. New York: ACM Press.

Resnick, P., and H. R. Varian, eds. 1997. Special section: Recommender systems. *Commun ACM* 40(3):56–89.

Rheingold, H. 2002. *Smart Mobs: The Next Social Revolution*. New York: Basic Books.

Robertson, G., M. Czerwinski, P. Baudisch, B. Meyers, D. Robbins, G. Smith, and D. Tan. 2005. The large—display user experience. *IEEE Comput Graph Appl* 25(4):44–51.

Rocco, E. 1998. Trust breaks down in electronic contexts but can be repaired by some initial face-to-face contact. In *Proceedings of CHI '98*, 496–502. Los Angeles, CA: ACM Press.

Rocco, E., T. Finholt, E. C. Hofer, and J. D. Herbsleb. 2000. *Designing as if Trust Mattered*. (CREW Technical Report). Ann Arbor, MI: University of Michigan.

Ruhleder, K., and B. Jordan. 2001. Co-constructing non-mutual realities: Delay-generated trouble in distributed interaction. *Comput Support Coop Work* 10(1):113–38.

Satzinger, J., and L. Olfman. 1992. A research program to assess user perceptions of group work support. In *Proceeding of CHI '92*, 99–106. Monterey, CA: ACM Press.

Schafer, W. A., C. H. Ganoe, and J. M. Carroll. 2007. Supporting community emergency management planning through a geocollaboration software architecture. *Comput Support Coop Work* 16:501–37.

Schafer, J. B., J. Konstan, and J. Riedl. 2001. Electronic commerce recommender applications. *J Data Min Knowledge Discovery* 5(1/2):115–52.

Schatz, B. R., and J. B. Hardin. 1994. NCSA Mosaic and the World Wide Web: Global hypermedia protocols for the Internet. *Science* 265:895–901.

Schmandt, C., and S. Marti. 2005. Active messenger: E-mail filtering and delivery in a heterogeneous network. *Hum Comput Interact* 20:163–94.

Schmidt, K. 2002. The problem with 'awareness.' *Comput Support Coop Work* 11:285–98.

Schmidt, K. 2010. 'Keep up the good work!': The concept of 'work' in CSCW. In *Proceedings of COOP 2010*, 265–85. London: Springer-Verlag.

Setlock, L. D., S. R. Fussell, and C. Neuwirth. 2004. Taking it out of context: Collaborating within and across cultures in face-to-face settings and via instant messaging. In *Proceedings of CSCW 2004*, 604–13. New York: ACM.

Short, J., E. Williams, and B. Christie. 1976. *The Social Psychology of Telecommunications*. New York: Wiley.

Siegel, J., V. Dubrovsky, S. Kiesler, and T. W. McGuire. 1986. Group processes in computer-mediated communication. *Organ Behav Hum Decis Process* 37(2):157–87.

Singhal, S., and M. Zyda. 1999. *Networked Virtual Environments: Design and Implementation*. New York: Addison-Wesley.

Sproull, L., and S. Kiesler. 1991. *Connections: New Ways of Working in the Networked Organization*. Cambridge, MA: MIT Press.

Starner, T., and B. Rhodes. 2004. Wearable computer. In *Berkshire Encyclopedia of Human-Computer Interaction*, ed. W. S. Bainbridge, Vol. 2, 797–802. Great Barrington, MA: Berkshire Publishing Group.

Straus, S. G. 1996. Getting a clue: The effects of communication media and information distribution on participation and performance in computer-mediated and face-to-face groups. *Small Group Res* 1:115–42.

Straus, S. G. 1997. Technology, group process, and group outcomes: Testing the connections in computer-mediated and face-to-face groups. *Hum Comput Interact* 12(3):227–66.

Straus, S. G., and J. E. McGrath. 1994. Does the medium matter: The interaction of task and technology on group performance and member reactions. *J Appl Psychol* 79:87–97.

Tanenbaum, A. S. 2011. Computer networks (5th Ed.). Boston: Pearson Education.

Tang, J. C., and E. Isaacs. 1993. Why do users like video? *Comput Support Coop Work* 1(3):163–96.

Tang, A., M. Tory, B. Po, P. Neumann, and S. Carpendale. 2006. Collaborative coupling over table top displays. In *Proceedings of CHI 2006*, 1181–90. New York: ACM.

Tang, J. C., N. Yankelovich, J. Begole, M. van Kleek, F. Li, and J. Bhalodia. 2001. ConNexus to Awarenex: Extending awareness to mobile users. In *Proceedings of CHI 2001*, 221–8. New York: ACM.

Teasley, S. D., L. A. Covi, M. S. Krishnan, and J. S. Olson. 2002. Rapid software development through team collocation. *IEEE Trans Software Eng* 28:671–83.

Veinott, E., J. S. Olson, G. M. Olson, and X. Fu. 1999. Video helps remote work: Speakers who need to negotiate common ground benefit from seeing each other. In *Proceedings of the Conference on Computer-Human Interaction, CHI '99*, 302–9. Pittsburgh, PA: ACM Press.

Vertegaal, R. 1999. The GAZE groupware system: Mediating joint attention in multiparty communication and collaboration. Paper presented at the CHI '99, New York: ACM Press.

Vertegaal, R., R. Slagter, G. van der Veer, and A. Nijholt. 2001. Eye gaze patterns in conversations: There is more to conversational agents than meets the eye. Paper presented at the CHI 2001, New York: ACM Press.

Viegas, F. B., M. Wattenberg, and K. Dave. 2004. Studying cooperation and conflict between authors with history flow visualizations. In *Proceedings of CHI 2004*, 575–82. New York: ACM.

Vieweg, S., A. Hughes, K. Starbird, and L. Palen. 2010. Microblogging during two natural hazard events: What Twitter may contribute to situational awareness. In *Proceedings of CHI 2010*, 1079–88. New York: ACM.

Wang, H.-C., S. R. Fussell, and L. D. Setlock. 2009. Cultural difference and adaptation of communication styles in computer-mediated group brainstorming. In *Proceedings of CHI 2009*, 669–78. New York: ACM.

Wellman, B. 2002. Little boxes, globalization, and networked individualism? In P. van den Besselaar & T. Ishida (eds.), *Digital Cities II: Computational and Sociological Approaches*, 10–25. Berlin: Springer.

Whittaker, S. 2005. Supporting collaborative task management in e-mail. *Hum Comput Interact* 20:49–88.

Whittaker, S., V. Bellotti, and P. Moody. 2005. Introduction to this special issue on revisiting and reinventing e-mail. *Hum Comput Interact* 10:1–9.

Whittaker, S., E. Geelhoed, and E. Robinson. 1993. Shared workspaces: How do they work and when are they useful? *Int J Man Mach Stud* 39(5):813–42.

Whittaker, S., and C. Sidner. 1996. E-mail overload: Exploring personal information management of e-mail. In *Proceeding of CHI '96*, 276–83. Vancouver, BC: ACM Press.

Williams, E. 1977. Experimental comparisons of face-to-face and mediated communication: A review. *Psychol Bull* 84:963–76.

Wilson, S., J. Galliers, and J. Fone. 2006. Not all sharing is equal: The impact of a large display on small group collaborative work. In *Proceedings of CSCW 2006*, 25–8. New York: ACM.

Wiltse, H., and J. Nichols. 2009. PlayByPlay: Collaborative web browsing for desktop and mobile devices. In *Proceedings of CHI 2009*, 1781–90. New York: ACM.

Winograd, T. 1988. A language/action perspective on the design of cooperative work. *Hum Comput Interact* 3:3–30.

Wong, A., and A. Young. 2009. *Network Infrastructure Security*. New York: Springer.

Yang, H., and G. M. Olson. 2002. Exploring collaborative navigation: The effect of perspectives on group performance. In *Proceedings of CVE '02*, 135–42. New York: ACM Press.

Zhao, D., and M. B. Rosson. 2009 How and why people Twitter: The role that micro-blogging plays in information communication at work. In *Proceedings of GROUP 2009*, 243–52. New York: ACM.

Zheng, J., N. Bos, J. S. Olson, D. Gergle, and G. M. Olson. 2001. Trust without touch: Jump-start trust with social chat. Paper presented at the Conference on Human Factors in Computing Systems CHI-2001. Seatle, WA: ACM Press.

25 Human–Computer Interaction and the Web

Helen Ashman, Declan Dagger, Tim Brailsford, James Goulding, Declan O'Sullivan, Jan-Felix Schmakeit, and Vincent Wade

CONTENTS

25.1 INTRODUCTION

As successful as the web is for delivering information globally and rapidly, many problems remain which make it a challenging or unproductive experience for some users, or even impossible for other users.

Pioneering research work in the early 1980s introduced and defined the concept of usability as critical to the success of interactive products and systems (Eason 1984). From the 1990s, there has been a developing relationship between usability and the web, such that for many people, the two terms are inextricably linked. Specifically, not only does the web demand good usability, it can be argued that usability necessitates the web. There are five key reasons why usability and the web have become so closely associated with each other:

1. *Web use is approaching ubiquity.* For many countries, the people who access the web are approaching the population as a whole. For some countries (e.g., Australia and New Zealand), it has been estimated that around 80% of the population are using the web, while the entire North American region has more than 76% of the population using the web.* It seems reasonable to predict that in the future, Internet use will be as near-ubiquitous as telephone use. Such diversity raises many issues related to interdependent variables such as age, gender, culture, disability, language abilities, computer skills and knowledge, domain skills and knowledge, and so on. In particular, considerable ongoing research considers the accessibility of the web, that is, how can websites be designed to account for the needs of as many people as possible (e.g., those with visual impairments) (Abascal and Nicolle 2001; Goble, Harper, and Stevens 2000).
2. *Web users are largely discretionary users.* Apart from the specific case of the work context and company intranets, web users generally do not have to use a particular site (or even the web at all) to achieve their goals. They have alternatives available to them (e.g., another website, making a phone call, visiting a shop), and if they experience usability problems, they do not necessarily have to struggle on or adapt to the poor interface. They are empowered to explore the existing options.
3. *Web usability problems have a clear relationship with sales* (Moss 2010; Nielsen 2008). For websites aiming to sell products or services, poor usability directly impacts on sales. If a user cannot find the product or relevant information, they are unlikely to continue in the transaction. In this case, it has been noted that usability is affecting the experience of a product before purchase (Nielson 2000). Consequently, the website information space must consider many of the issues (e.g., navigation, layout of items, labeling) relevant to the traditional design of physical space for shops.
4. *The web is evolving at a rapid pace.* Technical characteristics are continually changing in response to new application/task areas, facilitated by computer processing power and communication speeds, but also by recent advances in interaction mode. This has an immediate impact on the functionality available to web users (e.g., animations, videos), developments in user-interface design (e.g., clickable items, mouse-over navigation), as well as tools (e.g., cookies, plug-ins). More recently, there have been radical shifts in the capabilities of mobile devices such as smartphones. These changes have firstly seen web applications becoming situation-aware, knowing their geographic location and communicating with other devices (such as other smartphones, other devices,† or radio frequency identification [RFID]), with varying levels of autonomy. Second, we also have seen significant changes in the usual input methods, with touch-sensitive devices creating interfaces that users find attractive, although these come with their own usability issues (Nielsen 2010b). In contrast, users' skills, knowledge, and expectations are often slower to evolve, leading to an inevitable gulf between users (particularly infrequent users) and the web. This is particularly notable when viewing the web on mobile devices (Nielsen 2010a).
5. *Website technical development is easy.* It is simple to have a presence on the web when compared with the resources necessary for traditional product-development processes. To generate a website requires few technical skills, and one can do so with no programming or usability experience, although clearly the basic capabilities do not necessarily make for a useable website. This has contributed to the vast number of websites, ultimately adding to the complications of navigation, while emphasizing the importance of usability as a differentiating factor.

In this chapter, we first consider what it is that makes the web hard to use. We look specifically at browsing and linking, finding things with search-and-query perhaps aided by semantic web classification, increasing the relevance of things we find with personalization and portals, communities and social networks, and finally the mobile web.

25.2 WHAT MAKES THE WEB HARD TO USE?

Much of the difficulty in using the web lies in the vast quantity of information available—browsing and searching becomes increasingly difficult to do and imprecise in its results. Results can be ambiguous, or quite general, and

* Figures for 2009 from www.internetworldstats.com.

† Even toys such as the AR drone (see http://ardrone.parrot.com/) interact with smartphones and have a programmable interface, which could make them largely autonomous of human drivers.

frequently there can be many millions of them. Then having located what one wants, technical flaws, such as broken links or browser incompatibilities often render the information unreachable or unreadable. Most importantly, web pages or the browser interface itself can give a poor presentation, and for some classes of users, make the information inaccessible.

In this section, we look at some of the problems with using the web. In subsequent sections, we look at the further developments in the web that address these issues.

25.2.1 Browsing and Linking: "What's Wrong with the World Wide Web" Revisited

Around 15 years ago, the difficulties in using the web were explored in depth, and presented a vision of "fourth-generation hypermedia" (Bieber et al. 1997). In particular, the importance of hypermedia in the structuring of information was described:

> Hypermedia provides contextual, navigational access for viewing information and … represents knowledge in a form relatively close to the cognitive organizational structures that people use. Thus hypermedia supports understanding.

The authors went on to define a table of desired hypermedia features (Bieber et al. 1997). Of particular relevance to human–computer interaction (HCI) and the web are the personalization of links, both by annotation and computation, and overviews.

Many of these desired features are still missing from the mainstream web applications. In particular, the generally applicable personalization of links and content is still largely unachievable without specialist tools. In fact, some backsliding on this is evident, as the mid-1990s Mosaic browser supported the creation of annotations at both personal and group level and, more importantly, offered this service by default in a standard browser.* Mosaic development, however, was discontinued when Netscape became widespread. W3C's Annotea project† offered the same essential functionality but not within a standard browser and not without setting up effort from a technically-literate user—annotations are by no means a standard or easily accessible web service.

Creating personalized annotations is challenging enough, but the creation of one's own links over third-party data sources is essentially unknown. This is not because of any shortcomings in the technology, as in fact many solutions have been available within the research community for some time (see Section 25.3). The earliest plans for web browsers, dating back to 1991 (Cailliau and Ashman 1999), anticipated a "writeable web" where personalized links and annotations would be easily supported, yet this has not become mainstream web technology.

Another key aspect of personalized links which is often overlooked is that personal links are private links. If one creates a web page, complete with links, then those links are necessarily as available as the content. This is not always desirable, especially if links record one's private associations and thoughts, and the intellectual property represented by those associations has value. The technology for personalized links gives the user private links, over public documents, in much the same way that users' bookmarks are private. For privacy reasons alone, the technology for personalized links should be requisite for all mainstream browsers.

Other desired hypermedia features such as local overviews are provided not as a standard web service but as a site-specific courtesy page optionally supplied by site managers. Global overviews seem an increasingly distant feature, as the vast complexity of the web makes the usual, graph-based representations completely unworkable, both computationally and visually. The earlier paper noted that,

> Web browsers have no inherent way of presenting the structure and interrelationships of data of any sort. For example, there is no way to visualize even the simple interrelationships of web documents, such as "Where can I go from here?" or "Which documents point to this document?" The reader has no idea of the position of a given document within the corpora unless an author explicitly embeds such details. (Bieber et al. 1997)

So why is it that so little has seemingly been done to address these outstanding problems? It could be because the technology is too inaccessible to the nonspecialist user and that the obvious upfront costs of assimilating the technology outweigh the perceived benefits. It still seems to be true that,

> the use of Web technology is in part determined by its capabilities. Readers learn to make do with the available tools instead of demanding better tools, perhaps because they are not aware of how better tools might help them. (Bieber et al. 1997)

Only when we make the web more usable to the nonspecialist user will the use of all but the most basic tools become commonplace, resulting in higher productivity and better assimilation of information in the user.

25.2.2 Finding Things on the Web

25.2.2.1 Search and Query on the Web

One recent estimate has placed the size of the World Wide Web at around 24 billion pages.‡ With this wealth of information, the web would be untenable without mechanisms to assist navigation and file location. The most common web tool in use today is the search engine. To use a search engine, the user must submit a series of terms known as a query. This query in some way formulates and embodies what the user of the search system would like to retrieve information about. The usual output produced by the search engine is a small set of links to a selection of web documents extracted from the billions available. These web pages represent the search engine's response

* In 1994, the Mosaic browser was the "killer application" for the Internet, being the first major web browser, freely available from the National Center for Supercomputing Applications (NCSA). Its developers later spun off the Netscape company.

† http://www.w3.org/2001/Annotea/.

‡ http://www.worldwidewebsize.com/

to the user's information need, each document having been determined by the system as probably relevant to the query.

Presented with a set of results, the user selects a web page. If the document fails to meet the user's requirement, they return to the search engine for another result, or modify the query in light of previous results. This continues until the user's need is satisfied or the user's commitment to the search wanes (Harman 1992a).

From the perspective of a typical user, this has two main problem areas:

1. The confusion and uncertainty surrounding query formulation
2. The impenetrability of seemingly endless results

In Section 25.4, we discuss these obstacles and review a range of solutions.

25.2.2.2 Relevance

Although browsing and search-and-query are excellent tools for finding information on the web, they remain generic functions whose behavior is the same, regardless of the context. What remains to be considered now is how to further filter the mass of information on the web according to its relevance, not to a query, but to the individual user or to a community of users.

Information systems are often designed for a hypothetical "average" user. This "one-size-fits-all" approach ignores diversity in cultural and educational backgrounds, abilities, objectives, and aspirations. An information system with a single-user interface for all users is conceptually the same as a car manufacturer selling a car in only a single color—"any color so long as it's black."*

One solution is to build personalizable information systems, delivering content specific to requirements of different users. Without such systems, effective universal accessibility to information cannot be achieved. With it, every computer user and special-interest group will have personally tailored access to information sources. Personalization can take many forms, and it can involve the tailoring of hypertext links, the tailoring of presentation (often an accessibility issue), and the tailoring of content.

Content personalization has many benefits, for not only users, either as individuals or within a community, but also for the providers of information. For individuals, the tailoring of content can reflect a wide variety of requirements, but is always aimed at providing information most pertinent to that user in their current context. In e-Learning applications, for example, more challenging lessons are not served to the user until mastery of prerequisite material is achieved. In e-Commerce, a simple form of user consensus underlies a recommender system that personalizes suggestions for further purchases based on the current users' purchasing history similarity to that of other users.

For groups, personalization of content could help create communities of common interest among otherwise disparate users. Personalization can also support existing communities or social networks of users, by creating portals for accessing materials which are "personalized" to the interests of the group or community, either with a recommender systems approach or by sharing of explicit personal recommendations.

Personalization of information can also improve the whole computer use experience for the two major groups of under-represented users: (1) non-American/English cultural and language groups, and (2) people with special information needs. The web may have been "invented" by Europeans (Cailliau and Ashman 1999), but its subsequent development from 1993 was based on United States and has a distinctly American culture, including a predominance of the English language, linguistic form and alphabets.† However, there are now programs of research aiming to "localize" content, both in information systems and in the web, so that content can be delivered according to cultural, linguistic, and location-specific needs.‡ There are also devices which are now able to seek out location-specific information, as described in Section 25.8.

For information providers and publishers, personalization could create a delivery of information as suitable as possible to each individual user or special-interest group. In the normal publishing model, it is not economically feasible for publishers or information owners to publish information in a multitude of different forms, but with personalization and adaptation (see Section 25.6), it would be possible to do the personalization "on the fly" at no additional cost to the publishers, bringing enormous economic advantage to publishers using the technology.

This is essentially the same principle encountered in teaching—lectures present the information in a fairly general fashion, whereas tutorials or other small-group teaching give the students a chance to specify exactly what they need to know and to be given examples most helpful to them. In fact, the pressure on teaching resources has partly motivated adaptation and personalization in e-learning, and similarly it is the lack of resources preventing publishers from delivering their materials tailored to special-interest groups and to the individual. With e-learning, adaptation and personalization have arisen because teachers must offer personalized tutoring but are increasingly short of resources. With e-commerce, multicultural groups and special needs groups, adaptation and personalization can also be applied to offer an equally valuable specialization of information that is just not otherwise available.

25.2.3 User Interface Issues

Context played an important part in the personalization of content and links. However, context also needs to be

* Henry Ford is credited with saying "You can paint it any color, so long as it's black," but the Model T eventually appeared in many colors. Customer demand can motivate personalization.

† The American culture and set of assumptions is reflected in not only issues such as inadvertently rendering Arabic script backwards (left to right) because browsers render left to right by default, but also in American spelling of HTML tags, such as "color" and "gray."

‡ See, for example, http://www.cngl.ie/research.html.

considered in any environment, not just for personalization purposes.

25.2.3.1 Context of Use

When investigating the usability of web user interfaces, one must first consider the overriding issue of context. Context of use is seen as a critical constituent of usability, defined by ISO (1998) to "consist of the users, tasks and equipment (hardware, software and materials) and the physical and social environments in which a product is used." With respect to the web, the user, task, and environment issues are constantly evolving, leading to new challenges for HCI researchers and practitioners. For instance, the traditional environment for web users would have been the workplace, but statistics show that this is no longer the single situation for web use. Eighty-eight percent of U.K. web users access the Internet primarily from home.* Also, an increasing number of people now access the Internet through mobile devices (e.g., laptops, phones, personal digital assistants). In these situations, designers must consider a much wider range of physical environment factors, for instance, varying lighting, noise and thermal conditions, as well as other tasks that users may simultaneously carry out.

For example, there are widespread concerns regarding the use of Internet services within road-based vehicles (Lai et al. 2002; Burnett, Summerskill, and Porter 2004). Access to the web while driving may provide a range of tangible benefits to drivers, some driving related (e.g., prebooking a parking space, accessing real-time traffic information), and others oriented toward productivity or entertainment needs (e.g., viewing the latest information on stocks, downloading MP3 files). Traditional access to the web through desktop computers has utilized user-interface paradigms that are both highly visual and manual (e.g., scanning a page for a link and then using a mouse to point and then click on the link). Such activities are in clear conflict with the safety/time-critical task of driving which places heavy demands on the visual modality, while requiring continuous manual responses (e.g., turning the steering wheel). Clearly, designers need to establish fundamentally new interaction styles for use in a driving context of use (e.g., speech recognition, voice output, haptic interfaces). As part of this process, designers of interfaces for in-car Internet services must consider whether drivers should be given access to functionality while the vehicle is in motion. There is considerable ongoing research to provide guidance as to what constitutes an overly distracting interface (Lai et al. 2002).

25.2.3.2 Navigation Issues

At one time, it was asserted that the two biggest difficulties facing web users are download times and navigation (Nielson 2000; McCracken and Wolfe 2004). Download times are largely governed by the technical capabilities of Internet connections (which continue to increase as demand increase), together with clear design variables (e.g., the size of individual website pages). Navigation issues are considerably more complex. According to one diary-based study (Lazar et al. 2003), between one-third and one-half of time spent using a computer is unproductive, a situation predominately attributed to problems in web navigation.

In analyzing the navigation problem, we must first consider what is meant by the navigation task. A CHI workshop on this topic from 1997 (Jul and Furnas 1997) commented that a difficulty with research in the area was that authors tended to define navigation in varying ways. A broad view (adapted from the CHI workshop) is taken here, in which navigation can be said to involve the following:

- Planning "routes": When people navigate through space (whether real or virtual), they must first consider their overall strategy, that is, what methods will be appropriate in the current situation. For the web, a range of methods exist which aim to assist the user when deciding how to navigate across the web (e.g., search engines, directories, URLs) and/or within specific sites (e.g., site maps, navigation menus, links). Many people find it difficult to generate a suitable plan, for a range of reasons, either concerning basic cognitive limitations (such as remembering URLs), a lack of knowledge (choosing appropriate search terms, misunderstanding Boolean logic), or because methods are poorly implemented (e.g., confusing layouts for site maps).
- Following "routes": Once a high-level plan exists, people need to execute the subsequent point-by-point decisions, necessitating a virtual form of locomotion through the information space. In this stage, typical problems facing the web user often relate to the design of linking mechanisms between pages, for example, ambiguous link labeling, unclear graphics or icons, relevant information appearing offscreen, the need to visually scan large numbers of links, and so on.
- Orienting within the "space": The "where am I?" problem of the web is perhaps the one most discussed and researched (Otter and Johnson 2000). For optimum navigation performance and confidence, people need to have a sense of their current location in relation to their surroundings (e.g., their final destination, their start point, other key "landmarks," such as a home page). Orientation difficulties are often compounded on the web because users are "dropped" into a specific location that may not be the site designers' intended start point, through the use of search engines and bookmarks.
- Learning the "space": Repeated exposure to any large-scale environment (whether it be real or virtual) will lead to a deepening knowledge of objects within the space (e.g., particular pages), as well as an understanding of the various relationships between the objects (e.g., how pages follow each other, the overall structure of a site). These specific cases of

* Figures for July 2005 from www.statistics.gov.uk.

mental models are commonly known as cognitive maps, and facilitate fast and accurate navigation performance. Websites with poor differentiation (e.g., all pages appearing to be similar), low visual access (e.g., difficult to see where one can go next) and high path complexity (e.g., many links on a page) will all contribute to a poorly formed cognitive map (Kim and Hurtle 1995).

25.2.4 Summary

In the subsequent sections, we turn to some solutions for the problems outlined here, looking first at browsing and linking (Section 25.3), and then turning to searching and querying (Section 25.4), semantic web technologies (Section 25.5), personalizing the user experience (Section 25.6), communities and social web (Section 25.7), and the mobile web (Section 25.8).

25.3 BROWSING AND LINKING

The fundamental mechanism for viewing information on the web is by browsing implemented by hypermedia links.* However, this core capability still encounters technical problems which interrupt the user's browsing, such as difficulties in finding relevant material and limitations in the way relevant materials can be presented.

In this section, we look at some solutions, including automatic management of broken links, the easy personalization of links and how links can enable different perspectives on the same data.

25.3.1 Broken and Misdirected Links

Broken links, generally the well-known "error 404," remain an irritation for users of the web (Nielsen 1998). They are perceived to be "disruptive to the user experience" and a sign of an "unprofessional" website (Wikipedia 2010). However, broken links have frustrated users not only of the web but of earlier hypertext systems, and some of the first work on link integrity motivated the open hypermedia systems' principle of externalizing links, rather than embedding them into the data (Davis 1995).

Externalizing links implies that links are stored separately from the data being linked, so that reconciliation of the link to its referent was required, usually immediately before use, as a "late binding" of links (Brailsford 1999). However, any changes in the data meant that links could be either displaced, pointing to the wrong place in a document (part-of-file error, Davis 1998), or completely invalidated, with no document now known for the link to apply to (a whole-file error, Davis 1998).

HyperText Markup Language (HTML) links suffer from a similar problem, despite being embedded in the data because it is not only the source of the link (the "from" part of it) that must be reconciled to its referent, but also the destination (the "to" part). It is in the latter that HTML links are frequently wrong, with an estimated 23% of web links being broken within a year (Lawrence et al. 2001).

The solutions to broken links can be characterized as being *preventative* (creating infrastructure or procedures that avoid broken links), *corrective* (correcting broken links where they are discovered) or *adaptive* (never storing actual links, only instructions for making them as required) (Ashman 2000).

From the user's point of view, it might initially seem that preventative solutions are ideal because the irritation of broken or misdirected links will never happen. However, many of the preventative measures can only guarantee accurate links within a limited scope, and changes outside that scope (such as an entire domain name change) can still result in broken links. Also, they can be functionally limited; for example, it may be impossible to guarantee link integrity into information that is not part of the same preventative scheme.

Corrective solutions tend to be more robust, as they assume breakage will occur and have procedures in place to correct links, where possible, or to otherwise deal with them. These procedures are sometimes computations which aim to discover the new location for the linked document. These often function as mass correction procedures, taking place at intervals, which detect broken links and attempt to correct them, discard them, or at least to notify the page owner of the problem. From the everyday user's point of view, this is a reasonable form of solution, requiring little or no effort on their part, with breakages often not encountered by the user. However, it is still possible that the user will discover a broken link, increasingly so if it is some time since the most recent correction.

Also, the corrective approach may discard unfixable links that the user has previously required. This leaves the user with the knowledge that a link that was once present is now gone and seemingly unrecoverable. Perhaps a more user-friendly solution to irretrievably broken links are the so-called soft 404s (Bar-Yossef et al. 2004)—when pages go missing, those pages are replaced by human-readable error messages which essentially assume the identity of the missing page. They frequently offer the user the option of a search of the site, or perhaps redirect the user to a new location. It is estimated that 25% or more of all dead links are these soft 404s (Bar-Yossef et al. 2004).

The correction of links is very much a research problem still, with much effort expended into double-guessing what were the original link creator's intentions in making the link. However, if the link was originally created by a computation, then the correction often amounts to nothing more than executing the computation again. If for example, the link computation was to "link every instance of a person's name to their home page," then if the name is moved within various pages, or appears or disappears, the links can easily be reinstated to their correct positions.

* The very names of the two most important elements of the Web infrastructure, *hypertext* transfer protocol (http) and HTML indicate this clearly. The PageRank algorithm is based on link analysis, showing that links are an essential component in search engine indices as well.

The adaptive approach assumes that all data is subject to change, as do the corrective solutions. Instead of having procedures for correction in place, it guarantees correct links by never having links in storage. The instructions to create links are stored, and links are created as required from them. This dynamic approach was trialed in experimental systems, which created links either on user demand (Davis, Knight, and Hall 1994) or as a background process (Verbyla and Ashman 1994).

The adaptive approach is now evident in everyday web usage, for example, with the use of scripts to create and serve web pages dynamically, often as a result of a database query, where links are calculated and inserted at the time of serving. However, it cannot assist with links created and maintained by other users, and neither can the preventative or corrective approaches.

In the end, the major obstacle to link integrity in the web is its anarchic nature. There is no central authority which can impose robust linking practice on the mass of users, indeed the failure of the error-tolerant naming solutions such as uniform resource names (URNs) to capture the imagination shows that it is not even possible to tempt users into good linking practices, let alone force them.* Even a direct plea from one of the joint creators of the web has been unable to achieve link robustness (Berners-Lee 1998). As long as users continue to create links without troubling to maintain them, other users will eventually encounter them as broken links. There are however numerous solutions that, combined, will help keep the problem at bay, so that users may experience almost 404-free browsing.

25.3.2 Personalizing Links

Many users at some stage want to create their own links or annotations, generally to record their own associations between data, or to ease everyday information access. In fact, easing information access is an obvious but frequently overlooked purpose for links—links function as a form of "user pull" of information, hiding the information while still making its presence known, but providing that information on request with the minimum of user effort.

Users have different needs, and an author of web pages cannot anticipate all such requirements, let alone provide them. Even if all the potentially useful links were provided, not only would users disagree on the value of the links, but the interface and performance of the browser would suffer. For example, not every user wants a dictionary link, which could give a basic definition of any word selected by the users. However, non-native speakers of a language could find such a link invaluable. Glossary links are essential to a reader not familiar with technical terms, but become intrusive to seasoned readers. In each case, the users want to be able to "switch on or off" links to reflect their own needs.

This is easy with current research technology. Even as far back as 1992 (Davis et al. 1992), the technology to provide links over data not owned by the user was available. It was even quite simple to create a form of computed link, for example to link every occurrence of a name with a bibliography. Within a short time, it even became feasible not just to exploit link computation to automatically create one's own links, but to even create one's own link computation specifications, so that exactly the right computation, pointing into the right data collections, could be easily linked at the user's wish (Verbyla and Ashman 1994)—an early precursor of the web services principle so widespread today.

The technology that supports personalization of links has not yet propagated into mainstream web browsers. Yet the different solutions have been trialed in a web context; for example, the Distributed Link Service enabled individuals to make their own private link sets or to contribute to their group's collective link sets (Carr et al. 1995). Even the creation of one's own link computation specifications was trialed in a web environment (Cawley et al. 1995).

It is possible that a growing awareness for the need for privacy in online actions may yet motivate users to demand an easy-to-use personalized link facility. Even personalized annotations are a minority facility, only feasible for the reasonably technically literate user. Being able to record one's personalized links enables a user to record his or her private associations and collections, often representing original intellectual ideas or commercially valuable information, without advertising it to all and sundry. As more and more of each user's online interactions and transactions are exposed to outside scrutiny, authorized or otherwise, the provision of a feature with the fortunate side-effect of offering private means for recording ideas ought to become a priority for future web browser development.

We discuss further in Section 25.6 on how the personalization of links is increasingly feasible in the adaptive web.

25.3.3 Transclusion

Transclusion has always been a key component of the vision of hypermedia, ever since its earliest days (Nelson 1965). The term was originally used to denote the inclusion, by referential addressing, of part of one document within another; although its usage has now expanded to encompass the presentation of data in a context other than the one originally intended. With one important exception, transclusion was not until relatively recently widely implemented on the web. That exception is the HTML <IMG> tag—which transcludes an image into the context of the document. The image itself is neither embedded within the document nor copied—it is transcluded. However, in recent years, this concept has generated a great deal of interest, and features in a number of areas of web technology, including data interchange (Wilde and Lowe 2002), online publishing (Krottmaier 2002), multimedia (Kolbitsch 2005), and education (Moore, Stepp, and Diday 2001). Most importantly, it is now implemented in Media Wiki, and as such is an integral part of Wikipedia.†

* As evidenced by the lack of recent activity on URNs at the World Wide Web Consortium's *Naming and Addressing* page at http://www.w3.org/Addressing/.

† http://en.wikipedia.org/wiki/Wikipedia:Transclusion.

Although transclusion is a seemingly simple idea, it has been pointed out by Goulding, Brailsford, and Ashman (2010) that there are actually two quite distinct aspects to transclusion, called *instance transclusion* and *identity transclusion*, which are used for very different purposes. Identity transclusion is used to represent a unique entity, such as an image, in multiple contexts. Therefore, a single image may be transcluded onto many different web pages. Instance transclusion is closer to the concept of instantiation that is familiar to object-oriented programmers—in that different instances of an entity may be represented in different contexts. For example, in a system that models the actors that have played "James Bond," each transclusion represents the same fictional secret agent, but each instance is completely distinct from the others: Sean Connery's Bond is very much a different individual to Roger Moore's, even though they are both based on the creation of "Ian Fleming," a property they both inherit from the original entity. There is a distinction between the two uses of the word transclusion—an instance transclusion does not share all its relations across all of its contexts, whereas an identity transclusion does because for the latter, only one entity is being modeled. This is important, and a lack of appreciation of this has in the past led to confusion among the designers of different transclusion implementations.

25.4 SEARCHING AND QUERYING

Browsing, enabled by hypermedia links, was the first technology for accessing information on the web. However, the enormous success of the web and its rapid uptake by millions of users rapidly rendered browsing alone a manifestly inadequate tool for information access. Searching has now become a major, perhaps *the* major, means of locating information on the web.*

However, accessing information using search engines presents difficulties. First, the user faces the problem of creating an accurate description of his or her requirement. Second, the user must make sense of the results that the search engine produces. Solutions to these difficulties are discussed in this section.

25.4.1 Query Formulation

It is accepted that the typical user will experience difficulties when attempting to formulate an effective query. Three significant factors stand out:

1. *Low user commitment.* Users are reluctant to provide information beyond the bare minimum. In fact, what emerges is that search engine use is characterized by short queries and limited interaction coupled with unreasonably high expectations (Jansen, Spink, and Saracevic 2000; Rolker and Kramer 1999; Kobayashi and Takeda 2000):

> Real people in the real world, doing real information seeking and in a hurry, use Web search engines and give 2-word queries to be run against billions of Web pages. We expect, and get, sub-second response time and we complain when there are no relevant Web pages in the top 10 presented to us. (Browne and Smeaton 2004)

   Users, it would appear, are chary of commitment. They expect very-high quality results from each and every search instance rather than subscribing to the notion of search as an iterative process.

2. *Uncertain information needs.* Users often have an incomplete understanding of their information need, and their initial need will frequently mature during, and in direct response to, the process of searching (Lancaster 1968):

> Searchers normally start out with an unrefined or vague information need which becomes more sharply focused as their search continues and exposure to information changes their information need. (Browne and Smeaton 2004)

   This initial lack of clarity with regard to the object of the search readily translates into imprecise query terms, which in turn begets irrelevant and disappointing results. Some users quit the searching process at a very early stage, confounded by this apparent "failure."

3. *Difficulties in expressing their need.* Users may not know the correct syntax to frame their queries, or the commands to interact with the search engine. They may know in general terms what web pages they wish to retrieve, but struggle to find the query terms most likely to identify them:

> Except in special circumstances, it is difficult for a user to ask an information retrieval system for what they want, because the user does not, in general, know what is available and does not know from what it has to be differentiated. (Card, Robertson, and Mackinlay 1991)

   This is not surprising, as the formulation of a "successful" query requires some awareness of information retrieval theory, relatively uncommon among typical users. Some of this theory can be guessed by observant users (who might notice, for instance, that query terms which frequently occur in web pages produce poor results), but the remaining users must persevere with an environment in which much remains unsaid or unknown, never knowing why their search failed, or indeed why it succeeded.

To counter some of the problems mentioned above, researchers working with search engines have developed

* Four of the top 10 websites globally are search engines; see http://www.alexa.com/topsites.

strategies designed to improve the quality of a submitted query. One of these strategies is relevance feedback (RF):

1. The user submits an initial query and the search engine serves results.
2. The user then identifies relevant and non-relevant web pages using associated checkboxes, clickable links, radio buttons, and so on. This action supplies the search engine with feedback.
3. The search engine then automatically modifies the original query in response to the feedback. This may involve adding search terms to the query, known as query expansion. It may also involve reweighting of the query, where information in the relevant and non-relevant set of documents is used to modify the importance of various query terms.
4. The modified query is run by the search engine and a new set of search results is shown to the user.
5. This process continues until the user's information need is satisfied.

This technique is particularly successful, and it has been repeatedly demonstrated that RF can improve the performance of a search engine at comparatively little cost to the user or the system (Ruthven and Lalmas 2003; Harman 1992b; although see Spink, Jansen, and Ozmultu (2000), for contrast). For example, Koenemann (1996) noted that the

> … availability and use of relevance feedback increased retrieval effectiveness; and increased opportunity for user interaction with and control of relevance feedback made the interactions more efficient and usable while maintaining or increasing effectiveness.

However, it has also been found that in order for RF to be effective, the user must be offered a meaningful dialog for participation. Ruthven conducted a series of experiments examining the performance of user-supplied feedback measured against an automated equivalent. Commenting on the relatively low performance of the users, he identified a failure in the infrastructure supporting the user's search:

> … simple term presentation interfaces are not sufficient in providing sufficient support and context to allow good query expansion decisions. Interfaces must support the identification of relationships between relevant material and suggested expansion terms and should support the development of good expansion strategies by the searcher. (Ruthven 2003)

In summary, many search engine users find the formulation of an effective query difficult. RF offers an effective solution, providing

- The users are willing to commit to several search iterations.
- A suitable and useable feedback interface is implemented.

25.4.2 Results List

The results generated by a web search engine will frequently not satisfy the user. Sometimes, this negative result will occur because the search engine cannot find any documents that match the user's information need. However, it is much more likely to happen because the user has submitted a very broad query, resulting in too many documents. This difficulty has been identified as the "abundance problem," occurring when "the number of pages that could reasonably be returned as relevant is far too large for a human user to digest" (Kleinberg 1999).

Many solutions to this challenge have been suggested, but three interesting approaches are as follows: (1) community-based ranking algorithms; (2) improved visual interfaces; and (3) document clustering. We now examine each approach in turn.

25.4.2.1 Community-Based Ranking Algorithms

A search engine aims to rank a set of web pages in order of the likelihood that they will be relevant to the user's information need, with the document most likely to be relevant appearing first. This likelihood of relevance is usually calculated using a statistical measure related to the occurrence of the query terms in the documents concerned. Known as term frequency, this measurement is traditionally normalized with respect to document length and multiplied by a measure reflecting the specificity of each term within the document collection (Sparck Jones 1972; Aizawa 2003; Salton, Yang, and Yu 1975).

Document rankings dependent upon term frequency represent a purely arithmetical evaluation of the web pages concerned. This evaluation can (and does) provide a useful approximation of the likelihood of relevance to the user, but is by no means an authoritative measure. The underlying assumption—that term frequency translates directly into relevance—is exactly that: an assumption. There is no guarantee that a web page that contains a high frequency of the query terms will be any more relevant than a second page containing a lower frequency of those same terms. The latter may have fewer of the important keywords, but might be far superior in other, less quantifiable respects (e.g., it may be more concise, better illustrated, superior in style, easier to read, more complete in its references, etc.).

One solution to this problem has been to supplement the rudimentary rankings that can be constructed through statistical observations with more sophisticated sources of information. This has led to the development of a general class of ranking algorithms which implement citation-based metrics for relevance scoring (Garfield 1972; Pinski and Narin 1976). In these algorithms, the relative importance of each web page is a function of the number of other web pages that link to it (Brin and Page 1998). As Kleinberg (1999) observed, these "in-links … encode a considerable amount of latent human judgment." Accordingly, a web page that is referenced by a good proportion of its peers is a natural candidate for high ranking.

25.4.2.2 Improved Visual Interfaces

Hearst (1997) has observed that "long lists littered with unwanted irrelevant material" represent an unwieldy and nonintuitive method for delivering search results to the user.

Rather than expending resources developing more sophisticated ranking algorithms, Hearst asserted that the answer to this problem lay in shifting

> … the user's mental load from slower, thought intensive processes such as reading to faster, perceptual processes such as pattern recognition. It is easier, for example, to compare bars in a graph than numbers in a list. Color is very useful for helping people quickly select one particular word or object from a sea of others.

She went on to speculate that in the near future, advanced search interfaces may abandon the 2D page metaphor altogether, adopting "alternatives that allow users to see information on the Web from several perspectives simultaneously." A software implementation based on this very idea, known as the information visualizer (Card, Robertson, and Mackinlay 1991), has proved surprisingly powerful—in one experiment an organizational hierarchy requiring 80 printed pages was displayed on just one 3D screen (Robertson, Mackinlay, and Card 1991). It seems clear that a visual tool for searching the web which helps the user to "see" a set of search results rather than just "read" them would have considerable utility.

25.4.2.3 Document Clustering and Click Data as Relevance Feedback

Another way of simplifying a set of search results is to generalize. The process of generalizing a set of results begins with identifying particular commonalities shared by some members of that set (e.g., use or nonuse of certain terms, subject matter, file type, etc.). Web pages determined to be "similar" in some way are then grouped together under a single category or heading, and the user is subsequently presented with several coherent clusters of web pages rather than the traditional list. Provided the number of clusters is relatively low, this technique quickly reduces the cognitive load of studying the results, allowing a user to "skim" rather than to read.

There are numerous clustering algorithms for aggregating like documents in this way. While search engines require a function that indicates the relevance of documents to the search term, clustering algorithms generally require a similarity function that indicate the mutual relevance of documents to each other.

The in-links of a web page might be used recognized by Brin and Page (1998) and recognized by Kleinberg (1999) as indicators of latent document human judgment, but there is another, more dynamic and far more populous dataset of implicit human judgment. This alternative form of implicit human judgment is clickthrough data, and the subset of clickthrough data which makes up coselection data.

Clickthrough data (or just click data) is the implicit RF provided by users when they select a result from the results list of a search. Click data implies a measure of perceived relevance of selected URL to the search term. Coselection occurs where the user makes two or more clicks from a set of search results, implying in addition that the selected URLS are mutually relevant, so that coselections are a form of RF from URL to URL. This means that coselection data can be used as a similarity function for clustering URLs.

Click data has been considered as a form of relevance ranking, and while some research show they are not entirely reliable, at least over traditional, text-based search, other research show that they improve the relevance of results from image searches (Ashman et al. 2009). Furthermore, coselection data can further improve search results by being able to cluster search results into sense-singular aggregations; that is, it can be used to disambiguate a search term.

An example of coselection use was applied to the problem of ambiguity in query-based search (Truran, Goulding, and Ashman 2005). Lexical ambiguity causes considerable problems, complicating the already difficult task of establishing what a user is actually looking for, especially with the tendency to use few words in any query. An analysis of the query log of a very-large online encyclopedia recently found that 1 in 20 queries were largely wasted simply because a user chooses ambiguous query terms (Wen, Nie, and Zhang 2001). Truran, Goulding, and Ashman (2005) showed that clustering with coselections as a similarity function was capable of autocategorizing a set of search results into believable sense categories without any semantic analysis of the web pages concerned. Neither was an external knowledge source, such as a dictionary or expert system used. Instead, sense discrimination was the quiet background product of user consensus—the system studied and learned from its clientele, utilizing what Fitzpatrick and Dent (1997) termed the "information consuming habits" of its users to create a human-intuitive overview of the distinguishable semantics of the query, like documents grouped with like (Michalski, Stepp, and Diday 1983).

25.4.3 Summary

Users of a search engine often become impatient with the number of results returned when a query is submitted, and rarely proceed beyond the first page of suggestions (Harman 1992a). To ensure the most relevant documents appear on the first page, more sophisticated ranking algorithms have been developed which exploit the inherently self-referential nature of the web.

However, even with these improvements, search engines are still serving large chunks of unrefined information to their users. This seems likely to change soon. Search engines will almost certainly begin to abstract categories and significant groupings from any given set of results automatically, and visual interfaces that simplify the user's task seem probable.

25.5 SEMANTIC WEB

The rapid rise in the usage and sophistication of web content and services has led to the desire to make such content and services more discoverable and usable by applications at runtime. A key problem in meeting this desire is how to expose the semantics (meaning) of the web content or web service interfaces in a manner that is widely interpretable by applications. Significant progress has been made in addressing this problem over the last number of years, with proposals to express the semantics of information and web service interfaces using languages based on the eXtensible Markup Language (XML). This section briefly introduces the opportunity that the vision encapsulates, the reality and progress that has been made with the underlying technologies, and argues that to achieve the vision, more focus is required with respect to human interaction issues.

25.5.1 Vision and Opportunity

In their seminal article in 2001, Berners-Lee, Hendler, and Lassila (2001) captured the vision of the semantic web and exposed to a wider audience the research that had been ongoing into semantic-based technologies over the previous years, a momentum which has continued over the past decade. The case study in the article involved the setting up by a pair of working children of a series of therapist appointments for an elderly parent at times that would suit the children's calendars. It envisioned how this would be achieved with little interaction and fuss through a combination of intelligent applications working in conjunction with semantically exposed web content and services.

At the core of the approach is the exposition of metadata related to a web resource or web service that refers to a semantic model that can be processed by an intelligent application to find out more about the metadata and how it relates to other concepts or models. For the semantic model, although several alternatives have been proffered (e.g., Web Service Modeling Ontology, XML Topic Maps [XTM 2001]), by far the most researched, advocated, and used is the Resource Description Framework (RDF)-based family of standards (Resource Description Framework [RDF], RDF Schema [RDFS], Web Ontology Language [OWL] [W3C Semantic Web Standards 2010]) that represent graphs of semantic information as triples. Progressively more formal semantics are attached to the nodes and associations within the graphs as one moves from the RDF representation, through RDFS, to OWL. So, for example, an association between two nodes can be typed as a "subclass" within all representations, but OWL gives the ability to declare an association as "symmetric." These semantic model representations are accompanied by a range of reasoners, which have been designed to support applications in interpreting the represented models consistently, independently of who produced the model or the type of reasoner used. A reasoner is defined as a system that allows one to infer implicitly represented knowledge from the knowledge that is explicitly expressed (Baader et al. 2007). So, for example, if in a model we might declare the association "friend_of" as symmetric, and we explicitly associate Bob as "friend_of" Alice, then a reasoner will infer that Alice is a "friend_of" Bob implicitly, even if not explicitly declared. There are a wide range of OWL reasoners that have been developed. Examples include KAON2 (Hustadt, Motik, and Sattler 2004), Pellet (Sirin et al. 2007), F-OWL (Zhou et al. 2004), and OWLPrime (Wu et al. 2008). Each implementation has its own functional and nonfunctional tradeoffs, including semantic expressiveness, computational complexity, memory footprint, and processor load. Although applications can embed these reasoners directly and manipulate the semantic models using XML technologies, typically manipulation of the semantic models and access to reasoner capabilities by applications has been facilitated through an API such as Jena (Carroll et al. 2004).

There is no shortage of case studies and use cases that describe the opportunity and breadth of the application of semantic technologies (W3C Examples 2010), but broadly they fall into three categories: (1) applications that enable better search/information retrieval; (2) those that enable better personalization; and (3) those that enable easier information integration.

25.5.2 Reality and Progress

Despite promising results from the various case studies and use cases, the semantic web and related technologies has yet to become mainstream. In their 2006 article, Shadbolt, Berners-Lee, and Hall (2006) examined the status of the semantic web and proposed some reasons for the lack of viral uptake of the approach. Their conclusion was that despite clear evidence of maturity and utility of the approach and technologies, the semantic web will not become widespread unless a point is reached where "serendipitous reuse of data, your own and others', becomes possible." To this end, they advocated the widespread publication of data using RDF, and promotion of the use of emerging "lightweight" semantic models.

Today there is evidence that progress is being made in these directions. The "linked open data" initiative was launched in 2007 by W3C, with the intention of publishing a wide range of publically accessible information as RDF triples, and interlinking them (W3C Linked Data 2010). According to the W3C in April 2010, "collectively, the data sets consist of over 13.1 billion RDF triples, which are interlinked by around 142 million RDF links." Data on the linked open data cloud spans a diverse range of topics, data sets exist which relate to music, statistical data, movies, and much more. As for lightweight semantic models usage, there have been some successes in popular usage of a number of these. For example, the friend of a friend vocabulary describing persons, their activities, and their relations to other people and objects has become popular. Another example is the Semantically-Interlinked Online Communities vocabulary, which is increasingly popular for interconnecting social media such as blogs, forums, and mailing lists to each other.

Most significant however, is the momentum that is gaining behind the embedding of RDF in eXtensible HyperText Markup Language (XHTML) documents, through "RDF—in—attributes (RDFa)." RDFa is a W3C Recommendation (W3C RDFa) that adds a set of attribute level extensions to XHTML for embedding RDF-based metadata within web documents. Support for this standard has already been announced by Yahoo, Google, Facebook, and most recently Drupal. Opening up the option of referencing RDF models in existing web pages will naturally help in the mainstreaming of semantic technologies within the existing web community.

25.5.3 Human Interaction

Although the momentum is encouraging, it is clear that the mainstreaming of the semantic web approach will falter unless more focus is placed on how to widen beyond knowledge engineers, the creation, maintenance, and mapping of the underlying semantic models and their referencing. Typically tools for working with semantic models have either taken an approach where the semantic model is presented in a tree-like structure (e.g., Protégé 2010) or as a graph, either using simple visualization (e.g., OntoViz 2010) or more sophisticated presentations (e.g., OntoSphere [Bosca, Bomino, and Pellegrino 2005]). Although both types of tools have been useful for knowledge engineers, up until a few years ago very few user trials had been undertaken, or very little consideration had been given to usability issues (Jameson 2006). A notable exception to this is the cognitive support underpinning and the usability evaluations undertaken as part of the development of the CogZ tool for knowledge engineers (Falconer and Storey 2007). However, over the last number of years, the research community has started to focus on the challenges involved in semantic model creation and management by people who are not knowledge engineers. Natural language interfaces have been developed to enable domain experts to get involved in the development of a semantic model (Dimitrova et al. 2008). Techniques have been developed to monitor and analyze a user's interaction with files on his or her computer and the user's interaction on the web, and allow the user to develop and maintain a semantic model of personal interests based on the information gathered, with a view to supporting more relevant and personalized information retrieval (Groza et al. 2007). Gaming approaches have been experimented with in order to engage ordinary users in semantic model creation and maintenance (Siopaes and Hepp 2008). An add-on to the Firefox web browser is being evaluated to see if ordinary web users will periodically engage in the mapping between semantic models, in this case a model representing RSS feeds mapping to a model representing the user's personal interests (Conroy et al. 2009).

Although such research initiatives are laudable, it is clear that significant effort is still required in tackling the challenges of human interaction by ordinary web users with semantic models, coupled with the continued progress in mainstream web community and tool support for RDF usage, before the promise of the semantic web vision will be fully realized.

25.6 PERSONALIZING THE USER EXPERIENCE

Web users regularly have difficulty finding exactly appropriate content for their needs. Yet, when one considers that users' searches (on Google or Bing), on average, only contain two or three keywords in their queries, the level of success of these search engines is relatively impressive. So what improvements can we make to both enhance the retrieval of content so that it is more suited to the user-intended task and circumstances? How can we enhance the presentation of the web content so that it can be more easily comprehended?

From the other perspective, content providers have traditionally adopted a "one-size-fits-all," which attempts to provide a "generalized view" of the content subject. Unfortunately, this can result in content which is not particularly suited to any one user group. Thus, how can we maximize the usefulness of the content provider's information so that it can be presented to the user in the most effective and relevant manner? How can we help content providers maintain content, which is intended for users with differing interests, prior knowledge, and goals? In this section, we will examine approaches of personalization, and community support that are becoming widely used to address these issues.

Web personalization attempts to dynamically present the web information in a form most easily utilized by a particular user (either based on prior knowledge of that user or explicit information provided by that user). Personalization empowers users by organizing or presenting content in ways which are more suited to that user's particular needs or circumstances. Personalization enables content (web) providers to focus on individual users' needs while maximizing the reuse of existing content.

Although personalization and general web portals have quite distinct approaches to web design, they are all aimed at producing an end user experience that is apparently uniquely appropriate for each individual. This is what might be called "my web." Personalization is where the contents of websites are presented (either transparently, or under user control), so that the material is appropriate for the needs and requirements of the individual. Personalization can be embedded within a website interface or offered through a web portal. Personalization can be achieved with a broad range of techniques of varying degrees of sophistication, from something as simple as a user-selectable color scheme through to an adaptive content management system based on a comprehensive user model. Examples of simple forms of personalization are commonly used for e-commerce websites; for example, recommending certain content or products, suggesting "more appropriate" products or services based on a user's location or origin.

Personalization can also be seen in web search portals or services. Web search services, which know a user's previous searches and identify that user's location/context, can dynamically (and transparently) influence the results retrieved so that more relevant content, sensitive to the context of that particular user, are presented. Search engines such as Google adopt some of these practices.

More sophisticated personalization is increasingly being seen in web application portals such as e-learning portals, tourist advisory portals, and information sites. This personalization can be at the level of individuals, communities, or organizations. Where personalization is operating at the level of community, there is an opportunity for the adapted experience to be shared by a community. The "community-based personalization" is becoming more important as the growth in user collaboration and instances of web communities become more prevalent.

Another way of understanding personalization is to think of it as involving three logical steps: (1) *adaptive retrieval* where the querying of the content is enhanced by knowing more about the users intention, goal, needs, or context; (2) *intelligent content reasoning* where the association of ideas/concepts in the subject area can be reasoned about and the related or relevant content identified; and (3) *adaptive content composition* where the content is actually assembled using smaller units of content to generate a webpage or a website where aspects of contents navigation, presentation, and interaction are all modified for the benefit of that particular user.

25.6.1 Personalization

Modern websites frequently consist of content dynamically drawn from back end databases and information sources. Thus the content displayed by a website is often dynamically retrieved and displayed. The question arises then, why retrieve the exact same information, composed the same way, for all users when we know users differ enormously? Using personalization, a website can prove to be more "motivating" to use and more "encouraging" for users to revisit.

Personalization began with simple forms of adaptation of web pages; for example, by displaying the actual user's name on the top of the web page and welcoming him or her back, or by providing a panel of suggested content in the home page which was potentially of specific interest to that user based on previous history using the web page. One of the longest established examples of this is http://www.amazon.com, which stores information about the customer's interests, gleaned from various sources, to generate a personalized home page and suggest items that are likely to be of interest. Adaptive hypermedia, now typically referred to as adaptive web, is an academic discipline that is dedicated to bringing personalization to the web (Brusilovsky 2007). The principal application areas of Adaptive Web systems have traditionally been in information kiosk-style systems, educational systems, and tourism. However, personalization is now emerging in areas as diverse as news access and publishing (Billsus and Pazzani 2007) to healthcare (Cawsey Grasso, and Paris 2007) and even within museum information systems (Brusilovsky and Maybury 2002).

Educational web-based systems in particular have progressed the use of personalization mainly because their aims were focused on making it easy for the users/learners to comprehend the content and be as contextually relevant as possible to the user (Conlan and Wade 2002). For example, traditional (face to face) education is naturally very adaptive. A student who does not understand a question will ask the teacher for an explanation. If the student does not understand the answer, then the teacher explains it in different terms—maybe simplifying, maybe using a more appropriate example, maybe using different modalities (e.g., pictures, video, etc.). Exactly how the teacher explains it depends on the needs and competences of the student, so the teacher adapts the content (in this case the explanation) to the individual student. The ultimate goal of adaptive educational systems is to provide this capability to web-based learning. So far, most such systems are mainly experimental, but they are important because the majority of adaptive web research has been applied in this area.

25.6.1.2 Adaptive to the User's Need or Adaptable by the User

A web system is said to be *adaptive* if it can perceive the needs of a particular user, and hence is usually transparent to the user (i.e., a user may not be aware of the personalization that is actually happening on her or his behalf). An *adaptable* web system relies on the user explicitly giving the system personal information and requires explicit intervention by the user to initiate the adaptation. Adaptive web systems base their content delivery on what the system perceives to be a user's need. Adaptable systems, however, use information obtained directly from the user—which in most cases will represent that user's desire. Because need and desire are not necessarily the same thing, there is potential for a conflict of interests. When considering adaptive systems, the underlying assumption is that the system knows better than the user what is most appropriate. Although this sounds disempowering (and if implemented badly it can be), there are many circumstances where this view is justified—particularly if the user does not understand the ramifications of a particular choice at a decision point.

For example, in educational systems learners may not always have all of the facts or the pedagogic background necessary to take control of their own teaching regime. In "real-life" education, this is why we have teachers rather than just resources. Implementing this "system knows best" paradigm without due care creates the potential for serious problems in an adaptive system. It is quite likely to result in frustration or anger if a user realizes that the system is going against his or her desires, or if the system's assumption of need is incorrect—particularly a problem with a poorly thought out or in appropriate user model. Well-designed adaptive systems do not take control away from the user, rather they provide suggestions (albeit sometimes strong ones) as to the most appropriate path. This is then a user interface issue, and it may take the form of reordering content or suggesting what content is considered most appropriate (e.g., a common mechanism is to change link color).

When we consider that many users query the web to perform different kinds of "informal learning," for example, to find out a fact or detail, to get an explanation of a concept or process, to inform themselves of information valuable to them, we can realize that personalization of both the retrieval and presentation of web information can greatly benefit the user.

25.6.1.3 Adaptive Web—Architectures and Models of Adaptation

The first generation of adaptive web systems tended to be monolithic systems in which the actual adaptation/adaptive behavior was embedded in the actual content itself (typically as HTML, scripted rules or some form of intelligent reasoning). Such adaptive systems tended to work for specific content as the rules were embedded in and written for that content. Developing another adaptive website usually entailed developing an entirely new adaptive system (and embedding new rules in the new content). Thus the effort of development was similar each time, with only the "know-how" being used across different developments. Authors of such adaptive websites needed to be proficient programmers and the adaptive websites tended to be very application-specific, such as AHA (version 1), ELM-ART, and Interbook.

The second-generation adaptive web architectures made a deliberate separation between the reasoning needed to decide the adaptive behavior/presentation and the actual content. This recognizes that there needed to be multiple information sources to decide the appropriate adaptive behavior and rendering, also that web content needed associated metadata which described the subject of the content, the form of the content, and so on. The rules about how this information could be used to generate a navigation or rendering of the content needed contained in other models. This approach was called the "multimodel approach" as it typically entailed a minimum of three models: (1) a user model capable of representing instantiations of different users; (2) a content model which was a metadata description of the content pages that were to be adaptively navigated or rendered; and (3) a narrative model which contained the rules that when executed generated new "navigations" over different subjects (which the content entailed) and different renderings (Conlan and Wade 2002). The narrative was somewhat independent of the content in that it reasoned about the subject or kind of content to be adaptively navigated rather than reasoning directly about the instances of those subjects (i.e., about the specific pages themselves). Thus the narrative generated a "navigation" across subjects which was then mapped down to specific web pages to be retrieved. The part of the adaptive system (typically called an adaptive engine or adaptive player) used this navigation across the subject(s) to choose appropriate web page instances and renderings. This enabled the adaptive engine to be independent of the actual content and so greatly reduced the cost of development of new adaptive websites or portals. Moreover, authoring an adaptive website or portal, only required the authoring of the user model, metadata models, and narrative models. Thus a programmer was no longer required to author new sites. Examples of such adaptive systems include APeLS (Conlan and Wade 2002) and AHA (De Bra, Smits, and Stash 2006).

The current generation of adaptive web systems has expanded further on the multimodel approach and includes the runtime reconciliation of multiple kinds of models, such as user models, context models, narrative models, environment models, and web service models to generate unique navigations and renderings of webpages. They are typically portal-based and examples include GRAPPLE (De Bra, Smits, and Stash 2006), Knowledge Sea (Ahn, Farzan, and Brusilovsky 2006), and AE (O'Keeffe and Wade 2009). Authoring tools for such systems are intended for subject experts rather than technical programmers (i.e., users who know about the subject area of the website and can graphically author the models). Such authoring systems hide the complexity of the inference rule specification by offering graphical ways in which to indicate where adaptation should take place and what kind of information should be used to determine that adaptation. We call such adaptive systems "multidimensional" in that they can reconcile multiple dimensions of a user or context on which to base the adaptation. Examples of different dimensions that adaptive systems can adapt upon include content descriptions, subject areas, service descriptions, context (of use), delivery environment, and intended user process or activity. It is not the case that models for each of these dimensions is required for each adaptive system, but rather for different adaptive systems or systems which are intended to deliver adaptation for particular purposes or situations, tend to draw on at least three or more of these dimensions (Wade 2009).

25.6.1.4 Means of Adapting Content and the User Experience

The two most common approaches to implementing adaptation on the web is at the level of either content (i.e., adaptive presentation) or linking (i.e., adaptive navigation support). Within each of these, there are a number of techniques used that are described briefly in this section but are reviewed in detail elsewhere (Brusilovsky 1996).

The majority of adaptive presentation operates on text, although in principle any media type may be adapted. Adaptation of modality is where the adaptation operates at the level of choices between different media types (e.g., text, speech, animation, etc.). Although this is likely to become more important in the future, multimedia adaptation is currently relatively rarely implemented (one example is Fagerjord 2005). However, this is likely to increase in web applications supporting disability access or "hands busy/eyes busy" web applications.

There are a number of ways in which the adaptive presentation of text has been implemented—usually by manipulating fragments of predefined text. Some systems add fragments to a standard minimum body of text; others remove fragments that the system deems inappropriate. When fragments are removed they may be completely hidden, or they may be merely dimmed. Another variant of fragment removal is to use "stretchtext" (a body of text is hidden behind a single word or phrase and expanded on request). Some systems instead of adding or removing fragments of text simply reorder the fragments. A major problem with all forms of adaptive text presentation is the impact that it has upon narrative flow. Different users will necessarily have different narratives, which is not a problem per se other than the fact that it is possible (or even likely) that a web page will no longer form a coherent whole if its text is manipulated. This problem may be ameliorated by writing text in the form of conceptually discrete "atoms," but that causes quite serious problems with authoring (most people find it difficult to write following this sort of constraint) and makes legacy content difficult to adapt.

Adaptive navigation support has fewer problems with narrative flow because links are (usually) less fundamental to the sense of a page than is text. Adaptive linking has been divided into five main categories (Brusilovsky 1996): direct guidance, link sorting, link hiding, link annotation, and map adaptation. Direct guidance is where the system makes a recommendation as to the next node to visit, based on the adaptation criteria. This is often used for "guided tour" type of direction for novice users (i.e., it is tantamount to a linear sequence of "Next" buttons). Link sorting is where the ordering of links is changed—so that the most appropriate one is at the head of the list, or in a particularly prominent place. Link hiding is where inappropriate links are not displayed. This, of course, can make parts of the system inaccessible to some users, which may or may not be acceptable. A variant of link hiding is link dimming, where it is still possible to follow the link but is less prominent in the user interface than links that are deemed more relevant. In a link annotation system, visual clues are attached to links to give the user some idea of their status with respect to the system—for example, font sizes or colors can be used to provide a gradation of relevance. Lastly, in map adaptation systems a navigational map is generated according to the adaptation rules.

25.6.1.5 User Modeling

In order to be able to do any adaptation, adaptive systems need to know something about the person who is using them. This information is stored in a "user profile" either on the local (end user) machine or, far more commonly, in a centralized database (accessible by the adaptive system). In the user profile, as well as personal information (name, ID, etc.), the criteria that the system will use for adaptation are stored and these criteria constitute the user model. Although a detailed discussion of user modeling is beyond the scope of this chapter, the basic principles are very simple. A user model is simply a set of values that adaptive systems use to make decisions about relevance. One approach to user modeling is that of stereotype models (Kaplan, Fenwick, and Chen 1993). With these, users are grouped together with other like-minded individuals. For example, "2nd year Chemistry undergraduates" or "people who like cats" are both stereotypes and as such it is possible to make reasonable assumptions about members of those groups (e.g., "will know the structure of butane" or "will appreciate photos of cute kittens," respectively). The other main user modeling methodology is that of the overlay model (Fischer et al. 1990), where the individual's knowledge, background, and any other relevant data are overlaid onto the sum of all knowledge in the system. More frequently adaptive hypermedia systems adopt hybrids of these two approaches so that any given user is initially stereotyped, and as the system learns more about them, it uses an overlay model to record their progress (Zakaria and Brailsford 2002).

User modeling is critical for a usable adaptive system. If the user model is in some way flawed (i.e., if the information stored does not match the reality of users' background and behavior), then content is likely to be maladapted for a user, and the system is often extremely difficult to comprehend. This is compounded by the fact that there are no standards for user modeling, and most adaptive systems use their own proprietary user model. User Models in general should be "scrutable," meaning that the user can find out what information is being stored about him or her and why it is needed (Kay 2006). A growing concern is the privacy issues around eliciting and storing such user model information (Ashman et al. 2009).

25.6.1.6 Authoring

With third-generation adaptive systems, developing new adaptive information portals or experiences principally requires authoring of the adaptive models (e.g., user model, content model, narrative model, etc.) and ensuring relevant content is available. However, up until the relatively recently mature authoring systems for personalized, web systems were not available. However, some have begun to be used. Typically, these authoring tools are specific to a particular adaptive engine and application type. For example, the adaptive course construction tool was an e-learning authoring tool that enabled teachers to graphically represent the learning activities and concepts and to apply personalization based on different aspects of the learners, for example, competencies, learning goals, and so on (Dagger, Wade, and Conlan 2005). Other web-based adaptive e-learning authoring tools included My Online Teacher (Hendrix and Cristea 2007) and WHURLE (Meccawy et al. 2008).

However, there are a number of potentially difficult issues that need to be considered when attempting to design authoring interfaces for adaptive web-based systems (Cristea, Carro, and Garzotto 2005). First, there is a need for a new design paradigm, together with associated metaphors because personalization breaks the fundamental design paradigm of the web—that of the "page." When authoring a conventional web page, the author concentrates on content and can reasonably assume that users will see the page as it is created. Hence most authoring systems for static web pages use a fairly conventional WYSIWYG paradigm—quite similar to that of a word processor. In the case of adaptive systems, however, the authoring process requires a separation of content authoring from the authoring of the parameters for adaptation. Depending on the adaptation model that is used, what the author sees might be somewhat or completely different from what any given end user sees. Current authoring tools allow the author to graphically specify some potential "navigational" paths through the subjects, which the web content presents. The tools allow the author to indicate personalization criteria, for example, user interest, user prior knowledge that the authoring systems convert to adaptation rules. Thus the author need not write the rules themselves, but such rules are generated as a consequence of the author's identification of adaptive situations and the (user or context) information to be used in the adaptation.

Recently, there has been a move to ensure the authoring tools can create adaptive applications executable by different adaptive engines. A most recent attempt has been GRAPPLE project,* which has addressed this issue.

25.6.2 Recommenders

Another branch of personalization on the web has been the development of "Web Recommenders." These are similar in some ways to adaptive web systems, but specifically they suggest or recommend resources or web content to users. Simple examples are now quite common on the web, for example, Amazon's recommendations are based on previous book sales. There are typically three main categories of recommender systems: content-based, collaborative, and hybrid recommendation approaches (Adomavicius and Tuzhilin 2005). Content-based recommenders make recommendations based on the descriptions of the content and a profile of the user's interests. Collaborative (filtering)-based recommenders use the opinions of other users to filter or evaluate items and deduce their relevance to a user. Hybrid recommenders are recommenders which have combined the first two techniques along with other optimizations (Burke 2007). Recommender systems typically do not influence the actual composition or presentation of the recommended web resources and are therefore considered more as an enhanced web retrieval approach.

25.7 COMMUNITIES AND SOCIAL WEB

25.7.1 Communities

While an in-depth discussion of online communities is outside the scope of this section,† they can be described as "cyberspace[s] supported by computer-based information technology, centered upon communication and interaction of participants to generate member-driven content, resulting in a relationship being built" (Lee et al. 2003). From this definition, the makeup of an online community can be characterized as part technology, part sociology, and part psychology. This section will focus on the technologies. The idea of the Internet has always been rooted in communities, groups of people sharing common interests, goals, objectives, ownership and locale, and in creating links and bridging geographic, cultural, or temporal barriers. Part of the original vision of computer pioneers, such as Nelson (1982) was that people could easily share, edit, and comment on as well as read publicly available materials. Technological advances on the web are providing affordances to produce and share content and to network and interact at unprecedented rates. Web logs (more commonly referred to as blogs), of which there are more than 200 million,‡ allow a user to post material online which can be cited, rated, commented upon, and redistributed throughout the Internet. These blogs require no special knowledge of authoring or technical expertise and are usually accessed with a standard web browser.

Another technology that is important in the development of communities on the web is Wikis, which are community websites designed to facilitate collaborative authoring and knowledge sharing. The most spectacular example of this is the Wikipedia project—a vast encyclopedia written collaboratively by volunteers using a Wiki.§ The system provides facilities to discuss changes made to articles, and authors may vote on controversial issues—the editorial policy being the assumption that the exposure of material to many users will result in the addition of depth and elimination of errors. Wikipedia was founded in 2001, and by 2009, it contained more than 13 million articles, 78% of which were non-English.¶ However, these statistics change literally every minute.** While Blogs and Wikis, as relatively new community-enabling technologies, provide impressive statistics, the oldest and probably still largest community activity supporting technology is e-mail with an estimated 247 billion messages per day sent in 2009.††

However, as stated above, technology constitutes a very small part of an online community. The HCI design challenges lie in supporting the fabric, the governance and evolution of that community, and how the locus of control can be disseminated effectively among its users. For example, while central policy can be established to place boundaries on the types of interaction allowed (e.g., Wikipedia five pillars‡‡), the task of moderation should be driven by the ethics of the community members and their personally acceptable rules of interaction. These soft skills/interpersonal communication skills are a fundamental contributor to the longevity of online communities. The technologies chosen and the HCI principles applied should be driven by the lifecycle stage (inception, creation, growth, maturity, and death) of the online community (Iriberri and Leroy 2009).

25.7.2 Social Web

The term "Social Web" can be used to describe how people interact with people, how people interact with content, and how content interacts with content across the World Wide Web. These interactions can range from socializing and sharing information with friends, to playing games against peers, to planning and organizing events, to professional networking in your industry, to dating and relationship building, to seeking advice from others, to shopping, to traveling, and so forth.

While social networking sites to date largely focus interactions within their own silo (e.g., using Facebook, you can only primarily connect with other Facebook users), the Social Web looks in part to proliferate the interaction and connection affordances of social networking across the greater web. Similar to how first-generation web protocols enabled a user to jump between hypertext documents on the web, the Social

* www.grapple.eu.

† See Iriberri and Leroy (2009) for a detailed discussion.

‡ Wikipedia—http://www.wikipedia.org/.

§ Wikipedia—http://www.wikipedia.org/.

¶ http://socialnomics.net/2009/08/11/statistics-show-social-media-is-bigger-than-you-think/.

** http://en.wikipedia.org/wiki/Wikipedia:Statistics.

†† http://e-mail.about.com/od/e-mailtrivia/f/e-mails_per_day.htm.

‡‡ http://en.wikipedia.org/wiki/Wikipedia:Five_pillars.

Web is a move to enable the seamless transition, aggregation, and merging of social interactions online from many different sources, applications, and networks.

Many social network providers are moving toward this by lowering the barriers to interaction through "social plug-ins" or functional widgets that can be placed on any website to leverage some of the functionality of the source network (e.g., Twitter @anywhere and Facebook Like Button applications). Although their primary purpose is to drive more traffic to those networks, the spread of social interaction affordances across the web is gaining real traction (more than 1 million websites integrate with the Facebook platform*). Movements like the Federated Social Web† are addressing the technical and architectural challenges of a Social Web infrastructure by focusing on technologies to support and promote online identity and security, exchangeable and configurable profiles, definition, declaration and management of relationships, sharing of media and activity streams, messaging, indexing and searching, functional interoperability, and data portability. These enabling technologies will inevitably lead to a series of complex design challenges when building applications to leverage the capabilities and affordances of the Social Web.

The Social Web is bringing HCI online design practices to the forefront of web development. Design is no longer primarily focused on the user as a consumer. The user as a producer and a curator at both individual and community levels now takes center stage in the design process. Services like Facebook have set the bar so high for user interaction (usability or the behavior of a system in response to its users) and user experience (a subjective feeling of satisfaction with learning and using a system) that web users expect high levels of technical sophistication at the click of a button or the swipe of a finger. The Social Web is not seen as a virtual place where people congregate but as a ubiquitous medium that is becoming intertwined with our everyday, physical lives. People are accessing and producing social media on the go at exponentially growing rates‡ with increasing levels of associated context and augmented social experiences thanks to services like Foursquare and Scvngr.

The proliferation of social media production and sharing is sometimes referred to as the "Real-Time Web." As events happen around the world, whether big international events such as the World Cup or personal events such as "I met an old friend at the cinema last night," details of the event quickly make their way onto and across the Social Web through tweets, status updates, or some form of social media sharing. The more popular the event, the more network resonance (how the news of the event propagates across the Social Web) it creates. However, this ubiquity of information creates design problems of its own. Most importantly, how can we filter out all the noise that exists within the Social Web to get to the information that we want, when we want it? The "Right-Time Web" aims to leverage the methods of personalization described previously to adapt and tailor the information streams of the Social Web "just for me."

25.8 MOBILE WEB

25.8.1 Touch and Speech as Interaction Mechanisms

Touch-sensitive screen technology has been available since the late 1960s, and the ability to directly touch and manipulate information on a device is undoubtedly appealing to many users. Early touch screens were of very low resolution and could only sense a single touch, which made their applications quite crude—generally this involved pressing large hot spots or buttons. The ability for a touch screen to respond to multiple touches simultaneously, however, provides the possibility for control of interfaces by multifingered gestures (Wellner 1991), and such devices allow interactions that had previously used peripherals such as keyboards and mice to be easily accomplished by a single, highly space-efficient device (Westerman, Elias, and Hedge 2001). The phrase "multi-touch sensing" was coined by Han (2005) when he prototyped the technology that was later popularized by the Apple iPhone, and is now widespread in a large variety of smartphones, tablet computers, and other consumer devices. For some of these, such as the iPhone, the multitouch interface is the only means of interaction, although others (such as many Google Android devices) also contain a small physical keyboard.

The gestures used to interact with modern smartphones provide users with a rich new means of accessing web content, such as using finger-spreading movements to zoom and sliding movements to pan. This provides user interface designers with a toolset that goes a long way toward ameliorating the problems of displaying complex hypermedia content on small screens. However, the success and extremely rapid uptake of this technology has itself led to a number of design problems. Designers of different applications have quite frequently adopted their own nonstandard conventions and methods of interaction design. For example, a leftward swipe of a single finger might pan across an image or it might return to a previous web page. In both cases, this gesture represents a pan, but in the former example it is spatial, whereas in the latter, it is temporal (Wigdor, Fletcher, and Morrison 2009).

There are two major technologies that are used for touch screens, capacitive screens can only be operated by fingers and require a very light touch. Resistive screens may be operated by fingers, fingernails, or stylus and require a higher force threshold. Styluses enable more precise interaction than fingers alone, but they are inconvenient and easily lost. For text entry, some capacitive devices, such as iPhones, compensate for the loss of accuracy by the use of automated error correction. Physical feedback (i.e., sound or vibrations) helps to increase the usability of capacitive touch screens, and then for simple interactions, the user performance is similar, or even slightly superior to that of physical keys (Lee and Zhai 2009).

Speech recognition is now quite accurate, and in recent years, the technology has become reliable and cheap enough

[*] http://www.facebook.com/press/info.php?statistics.

[†] http://federatedsocialweb.net/wiki/Federated_Social_Web_Summit_2010.

[‡] http://mashable.com/2010/03/03/comscore-mobile-stats/.

to incorporate into a wide range of devices, and it is thus, technically at least, now a viable means of interaction. However, outside of specialized applications (mostly among blind or motor-impaired users), it has yet to become widely adopted. Part of the reason for this is undoubtedly simply the disruption that is caused by people speaking simultaneously in public places, and the fatigue that is caused by continuous speaking. However, Shneiderman (2000) has also pointed out some more fundamental problems with voice interfaces. Human–human voice interaction uses subtle variations of speed, intonation, and volume to convey a rich emotional subtext. Not only is this difficult to model in human–computer voice interaction systems, but it can actually disrupt the interaction. Moreover, the problem-solving parts of the brain also support verbal communication, so tasks requiring concentration are generally best carried out in silence—whereas physical tasks are handled by a different part of the brain. This means that most people find it easy to manipulate a touch or keyboard interface while thinking, but much harder to issue accurate voice commands. However, despite these problems, speech input is undoubtedly important for accessibility of smartphones. It is also possible that hybrid voice/touch interfaces might become important in the near future. An interesting experimental system of this type is the "earPod" developed by Zhao et al. (2007). This is an eyes-free menu system, which uses a touch interface for input and a speech system for output. Although the user performance was initially slower than with more conventional systems, after as little as 30 minutes practice, most users became very proficient with the system and could consistently outperform other user interfaces.

25.8.2 Situation-Aware Web-Enabled Devices

25.8.2.1 Context-Aware Devices

In recent years, mobile web devices, such as PDAs, smartphones and tablet computers, have led to the emergence of the mobile web as a direct extension of the "traditional" Internet. These devices, because of their increasingly smaller form-factor and intuitive touch-screen interface have opened a new way of accessing the web right from our fingertips. Users are no longer bound to a stationary computer or laptop to access the web; it is now available from an always-online, small device that fits into our pockets and allows access to the web at any time from any place. Services and applications are now becoming location-aware, using the physical location of the web device, and the context in which it is used, as an input to improve our interaction with these services (Virrantaus et al. 2006). This has led to many new opportunities in the area of mobile commerce (Rao and Minakakis 2003), mobile advertising, informational services, entertainment, education, or business processes (Varshney 2001). Although the first location-aware systems were developed in the 1990s, they still face many challenges today (Patterson, Muntz, and Pancake. 2003).

Various means of locating a mobile device have been developed and implemented, such as the first, true location-aware system based on radio transmitters and badges (Want et al. 1992). Work in related areas, such as augmented and virtual reality has also helped in refining location and position tracking in mobile devices. Optical sensors, based on infrared technology for example, were used by Ward et al. (1992). Schmalstieg and Wagner (2007) show the use of visual markers to provide orientation information on handheld devices. As Bluetooth technology has become available in mainstream mobile devices, it too has been utilized as a source of locality information (Aalto et al. 2004). RFID tags, based on low-powered radio-transmitters have also been used successfully, and Rashid et al. (2006), for example, created a real-world Pac-Man game, based on positioning information retrieved from RFID sensors.

However, global positioning system (GPS), a technology that relies on geostationary satellites to determine the location, is now used as the predominant means of locating a mobile device in the physical world today. It has been extended to the implementations found in modern devices, which rely on external processing to enhance positioning accuracy (Djuknic and Richton 2001), and have been used successfully in wide, open areas, but are inaccurate in indoor settings or areas without a clear view of the sky, a limitation of the consumer-grade GPS receivers included in such devices (Wing, Eklund, and Kellogg 2005). Benford et al. (2006) implemented an urban mixed-reality game, and Thomas et al. (2000) created an augmented reality version of the game Quake. These implementations show, however, that this technology can be used for many different applications.

The fundamental goal of location-awareness is fully context-aware systems (Harter et al. 2002) that adapt dynamically to the context in which they are used. The context of a system relates to its usage environment and is derived from sensory input (Hong, Suh, and Kim 2009), such as its physical location, its orientation, the current weather, time and date, and the status of the device. Services can then utilize this context to provide appropriate information, communication, or computation to its user. Ultimately, context-aware computing will pave the way for ubiquitous computing by fully integrating mobile systems into our everyday life.

25.8.2.2 Location-Aware Applications for Mobile Devices Today

Today, smartphones, such as the Apple iPhone or Android platform devices, include magnetic sensors for orientation information, and assisted GPS to provide location information. These devices are widely popular today, leading to the emergence of new applications and uses, such as location-awareness, to the general public.

However, even before the release and wide availability of mobile phones with positioning technology, Rao and Minakakis (2003) identified location-based services as a major step in the evolution of mobile commerce. The ability to pin-point a user's exact location has long been the dream of commercial users, for example, for mobile advertising, product or shopping information, entertainment, education, or the inclusion in business processes (Varshney 2001). End-users have also realized the opportunities location-aware devices could bring, as Kaasinen (2003) found in a user study, concluding that personalized, location-aware services can be beneficial and are accepted by users if they contribute positively to the overall experience.

Today, any application developed for the iPhone or Android platform can utilize location and orientation features. Google, for example, is utilizing the physical location of a device to supplement search results, while mapping and routing applications allow users to easily navigate the real world. Some other examples include Yelp,* providing location-aware crowdsourced reviews and recommendations, social networking (such as Google Latitude,† Foursquare,‡ and Gowalla§), crowdsourced traffic information, navigation and maps through Waze¶ as well as location-aware information services for weather and news.

The ever-increasing popularity of location-aware smartphones and applications have led to the emergence of urban computing, the integration of computing technology into our everyday life and daily activities (Kindberg, Chalmers, and Paulos 2007). This includes pedestrian navigation systems (Arikawa, Konomi, and Ohnishi 2007) and an adaption of Social Web applications to physical locations, utilizing 2D-barcodes, GPS, and blogs (Hansen and Grønbæk 2008).

25.8.2.3 New Interface to the Web: Augmented Reality

With the availability of orientation and positioning information in everyday devices, such as smartphones, augmented reality has become accessible to the general public.

Augmented reality refers to the "supplementation of the real world with virtual objects" (Azuma et al. 2001). Traditionally, mobile augmented reality systems have been developed based on specialized hardware and portable computers, military-grade GPS receivers for accurate location information and often required the user to wear a head-mounted display to provide an immersive environment (Hollerer et al. 1999; Thomas et al. 2000). However, as evaluated by Schmalstieg and Wagner (2008), utilizing small mobile devices, such as mobile phones, requires a different approach to the design of virtual environments. The "magic lens" style view has been developed for handheld devices, using the screen of the handset as a lens and augmenting the limited view available.

Papagiannakis, Singh, and Magnenat-Thalmann (2008) highlight the impact recent technological advancements of smartphones and availability of wireless wide area networks and 3G networks has had on the development of mobile augmented reality systems. Specialized hardware, weighing many kilograms, is not required anymore, as powerful smartphones can be used for this purpose instead.

Augmented reality systems require accurate location and sensor information to mesh the real-world view with additional, generated information. Implementations on smartphones and mobile handsets today rely on GPS location information, as well as gyroscopic sensory information to create augmented views. However, sensors available in smartphones are notoriously inaccurate and inconsistent, tainting the overall augmented reality experience (Wagner and Schmalstieg 2009). Further challenges include the social acceptance of systems, low-bandwidth wireless network infrastructure and limited graphical capabilities and memory of mobile devices. Inaccurate sensory information has been identified as a key problem in the augmented reality community, with work being currently underway to overcome these problems through the development of filtering and prediction algorithms, such as Gotow et al. (2010).

A number of specialized systems have already been developed that take advantage of this evolution of smartphones, such as Studierstube (Schmalstieg and Wagner 2008), as well as a new generation of mobile augmented reality games (Broll et al. 2008).

A number of applications are already available for the iPhone and Android platforms, such as Layar,** an augmented reality browser allowing users access to 3D informational overlays, and Wikitude,†† an augmented reality overlay for Wikipedia.

25.9 CONCLUSIONS

In this chapter, we have considered HCI in a specific web context, looking at the issues impeding smooth user interaction with web tools and documents. Although some problems remain unsolved, generally the usability of the web has improved greatly since its early days. We have also seen in the past few years an enormous expansion in the capacity of mobile devices, greatly broadening the reach of the web, as well as challenges posed by popular new input methods on context-aware devices.

REFERENCES

Aalto, L., N. Göthlin, J. Korhonen, and T. Ojala. 2004. Bluetooth and WAP push based location-aware mobile advertising system. In *Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services*, 49–58. New York: ACM.

Abascal, J., and C. A. Nicolle, eds. 2001. *Inclusive Design Guidelines for HCI*. Abingdon, UK: Taylor & Francis.

Adomavicius, G., and A. Tuzhilin. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans Know Data Eng* 17(6):734–49.

Ahn, J., R. Farzan, and P. Brusilovsky. 2006. Social search in the context of social navigation. *J Korean Soc Inf Manag* 23(2):147–65.

Aizawa, A. 2003. An information-theoretic perspective of tf-idf measures. *Inf Process Manag* 39(1):45–65.

* http://www.yelp.com.

† http://www.google.com/latitude.

‡ http://foursquare.com/.

§ http://gowalla.com/.

¶ http://world.waze.com/.

** http://www.layar.com/.

†† http://www.wikitude.org/.

Arikawa, M., S. Konomi, and K. Ohnishi. 2007. Navitime: Supporting pedestrian navigation in the real world. *IEEE Pervasive Comput* 6(3):21–9.

Ashman, H. 2000. Electronic document addressing - dealing with change. *ACM Computing Surv* 32(3):201–12.

Ashman, H., M. Antunovic, C. Donner, R. Frith, E. Rebelos, J.-F. Schmakeit, G. Smith, M. Truran. 2009. Are clickthroughs useful for image labelling? IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology 1:191–7.

Azuma, R., Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre. 2001. Recent advances in augmented reality. *IEEE Comput Graph Appl* 21(6):34–47.

Baader, F., D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider. 2007. *The Description Logic Handbook*, Cambridge University Press New York.

Bar-Yossef, Z., A. Broder, R. Kumar, and A. Tomkins. 2004. Sic transit gloria telae: Towards an understanding of the web's decay. In *Proc. WWW2004*, 328–37. New York: ACM.

Benford, S., A. Crabtree, M. Flintham, A. Drozd, R. Anastasi, M. Paxton, N. Tandavanitj, M. Adams, and J. Row-Farr. 2006. Can you see me now? *ACM Trans Comput Hum Interact* 13(1):100–33.

Berners-Lee, T. 1998. *Cool URIs Don't Change*. http://www.w3.org/Provider/Style/URI (accessed October 10, 2011).

Berners-Lee, T., H. Hendler, and O. Lassila. 2001. The semantic web. *Sci Am* 34–43.

Bieber, M., F. Vitali, H. Ashman, V. Balasubramanian, and H. Oinas-Kukkonen. 1997. Fourth generation hypermedia: Some missing links for the World Wide Web. *Int J Hum Comput Stud* 47(1):31–65. Special Issue on HCI and the Web. Academic Press.

Billsus D., and M. Pazzani. 2007. Adaptive news access. In *The Adaptive Web, Lecture Notes in Computer Science, LNCS*, vol. 4321, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl. Berlin, Heidelberg, New York: Springer-Verlag.

Bosca, A., D. Bomino, and P. Pellegrino. 2005. OntoSphere: More than a 3D ontology visualization tool. In *Proceedings of SWAP, CEUR. Workshop Proceedings, ISSN 1613-0073*, Trento, Italy.

Brailsford, D. 1999. Separable hyperstructure and delayed link binding. *ACM Comput Surv* 31(4es).

Brailsford, T., C. Stewart, M. Zakaria, and A. Moore. 2002. Autonavigation, links and narrative in an adaptive web-based integrated learning environment. In *Eleventh International World Wide Web Conference*.

Brin, S., and L. Page. 1998. The anatomy of a large-scale hypertextual web search engine. In *Proc. Seventh international Conference on World Wide Web (WWW7)*, 107–17.

Broll, W., I. Lindt, I. Herbst, J. Ohlenburg, A. Braun, and R. Wetzel. 2008. Toward next-gen mobile AR games. *IEEE Comput Graph Appl* 28(4):40–8.

Browne, P., and A. Smeaton. 2004. *Video information retrieval using objects and ostensive relevance feedback*. In *Proc. ACM symposium on Applied Computing*, 1084–90. New York: ACM.

Brusilovsky, P. 1996. Methods and techniques of adaptive hypermedia. *User Model User Adapt Interact* 6:87–129.

Brusilovsky, P. 2001a. Adaptive hypermedia. *User Model User Adapt Interact* 11(1/2):87–110.

Brusilovsky, P. 2001b. Adaptive educational hypermedia (invited talk). In *The 10th International PEG Conference*, 8–12.

Brusilovsky, P. 2007. Adaptive Navigation Support. In *The Adaptive Web, Lecture Notes in Computer Science, LNCS*, vol. 4321, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl. Berlin, Heidelberg, New York: Springer-Verlag.

Brusilovsky, P., and M. Maybury. 2002. From adaptive hypermedia to the adaptive web. *Commun ACM* 45(5):30–4.

Burke R. 2007. Hybrid web recommender systems. In *The Adaptive Web, Lecture Notes in Computer Science, LNCS*, vol. 4321, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl. Berlin, Heidelberg, New York: Springer-Verlag.

Burnett, G., S. Summerskill, and J. Porter. 2004. "On-the-move" destination entry for vehicle navigation systems – Unsafe by any means? *Behav Inf Technol* 23(4):265–72.

Cailliau, R., and H. Ashman. 1999. A History of Hypertext in the Web In *ACM Computing Surveys Special Issue on Hypertext and Hypermedia*, 31(4es). New York: ACM.

Card, S., G. Robertson, and J. Mackinlay. 1991. The information visualizer, an information workspace. *In Proc. SIGCHI Conf. on Human Factors in Computing Systems (CHI '91)*, 181–86. New York: ACM.

Carr, L., D. De Roure, W. Hall, and G. Hill. 1995. The distributed link Service: A tool for publishers, authors and readers. In *Proceedings of the 4th International WWW Conference*.

Carroll, J., I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. 2004. Jena: Implementing the semantic web recommendations. In *International World Wide Web Conference on Alternate Track Papers and Posters*.

Cawley, T., H. Ashman, G. Chase, M. Dalwood, S. Davis, and J. Verbyla. 1995. A link server for integrating the web with third-party applications. In *Proc. 1st Australasian World Wide Web Conf.*, Australia.

Cawsey, A., F. Grasso, and C. Paris. 2007. Adaptive information for consumers of healthcare. In *The Adaptive Web, Lecture Notes in Computer Science, LNCS*, vol. 4321, 465–84, ed. P. Brusilovsky, A. Kobsa, and W. Nejdl. Berlin Heidelberg New York: Springer-Verlag.

Conlan, O., and V. Wade. 2002. Multi-model, metadata-driven approach to adaptive hypermedia services for personalisation. In *the Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, AH2002*, 100–11. Berlin: Springer, LNCS 2347.

Conroy, C., R. Brennan, D. O'Sullivan, and D. Lewis. 2009. User evaluation study of a tagging approach to semantic mapping. In *European Semantic Web Conference, LNCS 5554*, 623–37. Berling: Springer.

Dagger, D., V. Wade, and O. Conlan. 2005. Personalisation for all: making adaptive course composition easy. Educ Technol Soc 8(3):9–25.

Davis, H. C. 1995. To Embed or not to Embed. *Commun ACM* 38(8):108–9.

Davis, H. C. 1998. Referential integrity of links in open hypermedia systems. In *Proceedings of Hypertext '98*, 207–16. New York: ACM.

Davis, H., W. Hall, I. Heath, G. Hill, and R. Wilkins. 1992. Towards an integrated information environment with open hypermedia systems. *In Proc. Second European Conference on Hypertext*, 181–90. New York: ACM.

Davis, H., S. Knight, and W. Hall. 1994. Light hypermedia link services: A study of third party application integration. In *Proc. ECHT '94*, 41–50. New York: ACM.

De Bra, P., D. Smits, and N. Stash. 2006. The design of AHA!, In *The Proceedings of the ACM Hypertext Conference*, 133, Odense, Denmark. New York: ACM.

Dimitrova, V., R. Denaux, G. Hart, C. Dolbear, I. Holt, and A. Cohn. 2008. Involving domain experts in authoring OWL ontologies. In *Proc. ISWC 2008, LCNS 5318*, 1–16. Berlin: Springer.

Djuknic, G., and R. Richton. 2001. Geolocation and assisted GPS. *Computer* 34(3):123–25.

Eason, K. 1984. Towards the experimental study of usability: Ergonomics of the user interface. *Behav Inf Technol* 3(2):133–43.

Fagerjord, A. 2005. Editing stretchfilm. In *Proceedings of the Sixteenth ACM Conference on Hypertext and Hypermedia*, 301. New York: ACM.

Falconer S., and M. Storey. 2007. A cognitive support framework for ontology mapping. In *Proc ISWC 2007, LNCS 4825*, 114–27. Berlin: Springer.

Fischer G., T. Mastaglio, B. Reeves, and J. Rieman. 1990. Minimalist explanations in knowledge-based systems. In *Proceedings 23rd Annual Hawaiian International Conference on System Sciences*, 309–17. Los Alamos, CA: IEEE Press.

Fitzpatrick, L., and M. Dent. 1997. Automatic feedback using past queries: Social searching? In *Proc. 20th Annual International Conf. on Res. and Dev. in Information Retrieval*, 306–13. New York: ACM.

Garfield, E. 1972. Citation analysis as a tool in journal evaluation. *Science* 178:471–79.

Goble, C., S. Harper, and R. Stevens. 2001. The travails of visually impaired web travelers. In *Proceedings of Hypertext 2000*, 1–10. New York: ACM.

Gotow, J., K. Zienkiewicz, J. White, and D. Schmidt. 2010. Addressing challenges in delivering augmented reality applications to smartphones. In *Proceedings of the Third International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications, Mobilware 2010*, Chicago, IL, USA. Brussels: ICST.

Goulding, J., T. Brailsford, and H. Ashman. 2010. Hyperorders and transclusion: Understanding dimensional hypertext. In *Proceedings of the 21st ACM Conference on Hypertext and Hypermedia*, 201–10. New York: ACM

Groza, T., S. Handschuh, K. Moeller, G. Grimnes, L. Sauermann, E. Minack, C. Mesnage, M. Jazayeri, G. Reif, and R. Gudjonsdottir. 2007. The NEPOMUK Project – On the way to the Social Semantic Desktop. In *Proc. of I-Semantics'07*, 201–11.

Han, J. Y. 2005. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, Proc. UIST '05*, 115–8. New York: ACM.

Hansen, F., and K. Grønbæk. 2008. Social web applications in the city: A lightweight infrastructure for urban computing. In *Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia*, 175–80. New York: ACM.

Harman, D. 1992a. Relevance feedback revisited. In *Proc. 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1–10. New York: ACM

Harman, D. 1992b. Relevance feedback and other query modification techniques. In *Information Retrieval, Data Structures and Algorithms*, ed. W. B. Frakes and R. Baeza-Yates, 241–63. NJ: Prentice-Hall, Inc.

Harter, A., Hopper, A., Steggles, P., Ward, A. and Webster, P., 2002. *The Anatomy of a Context-Aware Application*, Wireless Networks, 8 (2), 187–197.

Hearst, M. 1997. Interfaces for searching the web. *Sci Am* 276(3): 68–72.

Hendrix, M., and A. Cristea. 2007. A qualitative and quantitative evaluation of Adaptive authoring of Adaptive Hypermedia. In *Proceedings of the First European Conference on Technology Enhanced Learning, EC-TEL 2007*, 71–85. Crete, Greece: Springer LNCS (Doi): 10.1007/978-3-540-75195-3.

Hollerer T., S. Feiner, T. Terauchi, G. Rashid, and D. Hallaway. 1999. Exploring MARS: developing indoor and outdoor user interfaces to a mobile augmented reality system. *Comput Graph* 23:779–85.

Hong, J., E. Suh, and S. Kim. 2009. Context-aware systems: A literature review and classification *Expert Syst Appl* 36(4):8509–22.

Hustadt, U., M. Motik, and U. Sattler. 2004. Reducing SHIQ-description logic to disjunctive datalog programs. In *International Conference on Principles of Knowledge Representation and Reasoning*, 52–162. Danvers, MA: AAAI Press.

International Organisation for Standardisation. 1998. ISO 9241 Ergonomics requirements for work with visual display terminals (VDTs) – Part 11 Guidance on Usability.

Jameson, A. 2006. Usability and the semantic web. In *ESWC 2006, LNCS 4011*, 3. Heidelberg, Germany: Springer.

Jansen, B., A. Spink, and T. Saracevic. 2000. Real life, real users, and real needs: A study and analysis of user queries on the web. *Inf Process Manag* 36(2):207–27.

Jul, S., and G. Furnas. 1997. Navigation in electronic worlds: A CHI 97 workshop. *SIGCHI Bull* 29(4):44–9.

Kaasinen, E. 2003. *User needs for location-aware mobile services*. *Pers Ubiquitous Comput* 7(1):70–9.

Kaplan, C., J. Fenwick, and J. Chen. 1993. Adaptive hypertext navigation based on user goals and context. *User Model User Adapt Interact* 3(3):193–220.

Kay, J. 2006. Scrutable adaption: Because we can and must. In *the Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web Systems, Springer LNCS*, vol. 4018, 11–9. Dublin, Ireland. Berlin: Springer.

Kim, H., and S. Hirtle. 1995. Spatial metaphors and disorientation in hypertext browsing. *Behav Inf Technol* 14(4):239–50.

Kindberg, T., M. Chalmers, and E. Paulos. 2007. Guest Editors' Introduction: Urban Computing. *IEEE Pervasive Comput* 6(3):18–20.

Kleinberg, J. M. 1999. Authoritative sources in a hyperlinked environment. *J ACM* 46(5):604–32.

Kobayashi, M., and K. Takeda. 2000. Information retrieval on the web. *ACM Comput Surv* 32(2):144–73.

Koenemann, J. 1996. Supporting interactive information retrieval through relevance feedback. In *Conference Companion on Human Factors in Computing Systems*, 49–50. New York: ACM.

Kolbitsch, J. 2005. Fine-grained transclusions of multimedia documents in HTML. *J Univers Comput Sci* 11:926–43.

Krottmaier, H. 2002. Transcluded Documents: advantages of reusing document fragments. In *Proceedings of the 6th International ICCC/IFIP Conference on Electronic Publishing*, 359–67.

Lai, J., K. Cheng, P. Green, and O. Tsimhoni. 2002. On the road and on the web? Comprehension of synthetic and human speech while driving. In *Proceedings of SIGCHI 2002*, 206–12. New York: ACM.

Lancaster, F., 1968. *Information Retrieval Systems: Characteristics, Testing and Evaluation*. New York: John Wiley and Sons, Inc.

Lawrence, S., D. Pennock, G. Flake, R. Krovetz, F. Coetzee, E. Glover, F. Nielsen, A. Kruger, and C. Giles. 2001. Persistence of web references in scientific research. *IEEE Comput* 34(2):26–31.

Lazar, J., K. Bessiere, I. Ceaparu, J. Robinson, and B. Shneiderman. 2003. Help! I'm lost: User frustration in web navigation. *IT and Society* 1(3):18–26.

Lee, F., D. Vogel, and M. Limayem, M. 2003. Virtual community informatics: a review and research agenda. *Journal of Information Technology Theory and Application (JITTA)* 5(1):47–61.

Lee, S., and S. Zhai. 2009. The performance of touch screen soft buttons. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHI '09*, 309–18. New York: ACM.

McCracken, D., and R. Wolfe. 2004. *User-Centred Website Development – A Human-Computer Interaction Approach*. Upper Saddle River, NJ: Pearson Prentice Hall.

Meccawy, M., T. Brailsford, A. Moore, H. Ashman, and P. Blanchfield. 2008. WHURLE2.0: Adaptive Learning Meets Web 2.0. In *Proc. European Conf. on Technology-Enhanced Learning*. http://www.ectel08.org/.

Michalski, R., R. Stepp, and E. Diday. 1983. Automatic construction of classifications: conceptual clustering versus numerical taxonomy. *IEEE Trans Pattern Anal Mach Intell* 5:528–52.

Moore, A., T. Brailsford, and C. Stewart. 2001. Personally tailored teaching in WHURLE using conditional transclusion. In *Proceedings of the 12th ACM Conference on Hypertext and Hypermedia*, 163–64. New York: ACM.

Nelson, T. 1965. Complex information processing: a file structure for the complex, the changing and the indeterminate. In *Proceedings of the 1965 20th national conference, ACM '65*, 84–100. New York: ACM.

Nelson, T. 1982. *Literary Machines*. Watertown, MA: Eastgate Systems Inc.

Nielsen, J. 1998. *Fighting Linkrot*. http://www.useit.com/alertbox/980614.html (accessed July 5, 2010).

Nielsen, J. 2010a. *iPhone Apps Need Low Starting Hurdles*. http://www.useit.com/alertbox/mobile-apps-initial-use.html (accessed July 5, 2010).

Nielsen, J. 2010b. *iPad Usability: First Findings From User Testing*. http://www.useit.com/alertbox/ipad.html (accessed July 5, 2010).

O'Keeffe, I., and V. Wade. 2009. Personalised web: Adaptability for web service composition and web content. In *User Modeling, Adaptation, and Personalization: 17th International Conference, UMAP 2009*, Trento, Italy (LNCS 5535), ed. G.-J. Houben, G. McCalla, F. Pianesi, and M. Zancanaro, 480–86. Berlin: Springer-Verlag.

Ontology Visualisation tab for Protege. http://protegewiki.stanford.edu/wiki/OntoViz (accessed October 10, 2011).

Otter, M., and H. Johnson. 2000. Lost in hyperspace: metrics and mental models. *Interact Comput* 13:1–40.

Papagiannakis, G., G. Singh, and N. Magnenat-Thalmann. 2008. A survey of mobile and wireless technologies for augmented reality systems. *Computer Animat Virtual Worlds* 19(1):3–22.

Patterson, C., R. Muntz, and C. Pancake. 2003. Challenges in location-aware computing. *IEEE Pervasive Comput* 2(2):80–9.

Pinski, G., and F. Narin. 1976. Citation influence for journal aggregates of scientific publications: Theory, with application to the literature of physics. *Inf Proc Manag* 12:297–312.

Protégé. 2010. Ontology editing tool. http://protege.stanford.edu/ (accessed August, 2010).

Rao, B., and L. Minakakis. 2003. Evolution of mobile location-based services. *Commun ACM* 46(12):61–5.

Rashid, O., W. Bamford, P. Coulton, R. Edwards, and J. Scheible. 2006. PAC-LAN: mixed-reality gaming with RFID-enabled mobile phones. *Comput Entertain* 4(4):4

Robertson, G., J. Mackinlay, and S. Card. 1991. Cone Trees: Animated 3D visualizations of hierarchical information. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems, CHI '91*, 189–94. New York: ACM.

Rolker, C., and P. Kramer. 1999. Quality of service transferred to information retrieval: The adaptive information retrieval system. In *Proc. Eighth International Conference on Information and Knowledge Management*, 399–404. New York: ACM.

Ruthven, I. 2003. Re-examining the potential effectiveness of interactive query expansion. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '03*, 213–20. New York: ACM.

Ruthven, I., and M. Lalmas. 2003. A survey on the use of relevance feedback for information access systems. *Knowl Eng Rev* 18(2):95–145.

Salton, G., C. Yang, and C. Yu. 1975. A theory of term importance in automatic text analysis. *J Am Soc Inf Sci* 26(1):33–44.

Schmalstieg, D., and D. Wagner. 2007. Experiences with handheld augmented reality. *In Proc. 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, 1–13. Los Alamos, CA: IEEE Computer Society.

Schmalstieg, D., and D. Wagner. 2008. Mobile phones as a platform for augmented reality. *In Proc. IEEE VR 2008 Workshop on Software Engineering and Architectures for Realtime Interactive Systems*, 43–4. Washington, DC: IEEE Computer Society.

Shadbolt, N., T. Berners-Lee, and W. Hall. 2006. The semantic web revisited. *IEEE Intell Syst* 21(3):96–101. ISSN 1541–1672.

Shneiderman, B. 2000. The limits of speech recognition. *Commun ACM* 43(9):63–5.

Sirin, E., B. Parsia, B. Grau, A. Kalyanpur, and Y. Katz. 2007. Pellet: A practical OWL-DL reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web* 5(2):51–3.

Sparck Jones, K. 1972. A statistical interpretation of term specificity and its application in retrieval. *J Doc* 28(1):11–20.

Spink, A., B. Jansen, and H. Ozmultu. 2000. Use of query reformulation and relevance feedback by Excite users. *Internet Research: Electronic Networking Applications and Policy* 10(4):317–328.

Thomas, B., B. Close, J. Donoghue, J. Squires, P. Bondi, M. Morris, and W. Piekarski. 2000. *ARQuake: An outdoor/indoor augmented reality first person application*. In *Proc. 4th IEEE International Symposium on Wearable Computers*, 139. Washigton, DC: IEEE Computer Society.

Truran, M., J. Goulding, and H. Ashman. 2005. Co-active intelligence for information retrieval. In *Proceedings of ACM Multimedia '05*, 547–50. New York: ACM

Varshney, U. 2001. Location management support for mobile commerce applications. In *Proc. 1st International Workshop on Mobile Commerce*, 1–6. New York: ACM

Verbyla, J., and H. Ashman. 1994. A user-configurable hypermedia-based interface via the functional model of the link. *Hypermedia* 6(3):193–208.

Virrantaus, K., J. Markkula, A. Garmash, V. Terziyan, J. Veijalainen, A. Katanosov, and H. Tirri. 2006. Developing GIS-supported location-based services. In *Proc. Second International Conference on Web Information Systems Engineering*, 66–75. Washington, DC: IEEE Computer Society.

W3C, Examples. Semantic Web Case Studies and Use Cases. http://www.w3.org/2001/sw/sweo/public/UseCases/ (accessed October 10, 2011).

W3C, Linked Data–Connect Distributed Data across the Web. http://linked data.org/ (accessed October 10, 2011).

W3C, RDFa in XHTML:Syntax and Processing. http://www.w3.org/TR/rdfa-syntax/ (accessed October 10, 2011).

W3C, Semantic Web standards. http://www.w3.org/TR/ (accessed October 10, 2011).

Wade, V. 2009. Challenges for the multi-dimensional personalized web. In *User Modeling, Adaptation, and Personalization: 17th International Conference, UMAP 2009*, Trento, Italy, LNCS 5535, ed. G.-J. Houben, G. McCalla, F. Pianesi, and M. Zancanaro. Berlin: Springer-Verlag.

Wagner, D., and D. Schmalstieg. 2009. Making augmented reality practical on mobile phones, part 2. *IEEE Comput Graph Appl* 29(4):6–9.

Want, R., A. Hopper, V. Falcão, and J. Gibbons. 1992. The active badge location system. *ACM Trans Inf Syst* 10(1):91–102.

Ward, M., R. Azuma, R. Bennett, S. Gottschalk, and H. Fuchs. 1992. A demonstrated optical tracker with scalable work area for head-mounted display systems. In *Proc. Symposium on Interactive 3D Graphics*, 43–52. New York: ACM

Wellner, P. 1991. The DigitalDesk calculator: Tangible manipulation on a desk top display. In *Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology UIST '91*, 27–33.

Wen, J., J. Nie, and H. Zhang. 2001. Clustering user queries of a search engine. In *Proc. 10th Intl Conf. on World Wide Web*, 162–8. New York: ACM.

Westerman, W., J. Elias, and A. Hedge. 2001. Multi-touch: A new tactile 2-D gesture interface for human-computer interaction. In *Proceedings of the Human Factors and Ergonomics Society 45th Annual Meeting*, 632–36.

Wigdor, D., J. Fletcher, and G. Morrison. 2009. Designing user interfaces for multi-touch and gesture devices. In *Proceedings of the 27th International Conference Extended Abstracts on Human factors in Computing systems, CHI '09*, 2755–58. New York: ACM.

Wikipedia. 2010. Link Rot. http://en.wikipedia.org/wiki/Link_rot (accessed July 5, 2010).

Wilde, E., and D. Lowe. 2002. *XPath, XLink, XPointer, and XML: A Practical Guide to Web Hyperlinking and Transclusion*. Boston, MA: Addison-Wesley Professional.

Wing, M., A. Eklund, and L. Kellogg. 2005. Consumer-grade global positioning system (GPS) accuracy and reliability. *J For* 103:169–73.

WSMO. Web Services Modelling Ontology Technical Recommendations. http://www.wsmo.org/TR/ (accessed October 10, 2011).

Wu, Z., G. Eadon, S. Das, E. Chong, V. Kolovski, M. Annamalai, and J. Srinivasan. 2008. Implementing an inference engine for RDFS/OWL constructs and user-defined rules in Oracle. In *International Conference on Data Engineering*, 1239–48. Washington, DC: IEEE Computer Society.

XTM. 2001. XML Topic Maps. http://www.topicmaps.org/xtm/ (accessed October 10, 2011).

Zakaria, M., and T. Brailsford. 2002. User modelling and adaptive educational hypermedia frameworks for education. *New Rev Hypermedia Multimed* 8:83–97.

Zhao, S., P. Dragicevic, M. Chignell, R. Balakrishnan, and P. Baudisch. 2007. Earpod: eyes-free menu selection using touch input and reactive audio feedback. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1395–404. New York: ACM.

26 Human-Centered Design of Decision-Support Systems

Philip J. Smith, Roger Beatty, Caroline C. Hayes, Adam Larson, Norman D. Geddes, and Michael C. Dorneich

CONTENTS

26.1 OVERVIEW

Computers can assist decision makers in a variety of different ways. They can, for instance, support users with improved access to information, with more informative displays of this information, or with more effective forms of communication. They can also use algorithms to actively monitor situations and to generate inferences in order to assist with tasks such as planning, diagnosis and process control. This chapter focuses on interaction design issues (Preece, Rogers, and Sharp 2002; Scott, Roth, and Deutsch 2005) associated with this latter role, in which the software uses numerical computations and/or symbolic reasoning to serve as an active decision-support system (DSS) (Turban, Aronson, and Liang 2004).

An underlying assumption in design of human-centered systems is that the characteristics of the user population, the tasks they perform, and the context and environment in which they perform them are all inherently intertwined. DSS designers need to understand all three in order to understand what support a DSS should provide to help users to be more effective in a specific task and context. This chapter aims to provide DSS designers with approaches for understanding of all three, and their integration.

The chapter begins with a focus on design methods. We emphasize the value of an iterative design process with multiple cycles of design and testing with users, in increasingly realistic contexts. This process is strongly aided by the use of concrete scenarios and use cases to define the design problem, and the use of cognitive task analyses (CTAs) and work domain analyses to guide both the development of the underlying functions in the DSS, and the design of the interface that mediates the interactions of the user with these functions (Bittner 2002; Cockburn 2000; Kulak and Guiney 2003).

The chapter then proceeds to discuss aspects of human performance relevant to the design of DSSs. This includes a discussion of factors that limit human performance, as these factors suggest opportunities for improving performance through the introduction of a DSS. This discussion also includes consideration of the influence of a DSS on the user's cognitive processes, emphasizing the need to think in terms of the resultant joint cognitive system that is not a simple combination of the capabilities of the two agents (user and DSS) alone (Hoc 2001; Hollnagel and Woods 2005; Jones and Jacobs 2000).

The discussion further emphasizes the need to consider more than just the surface interactions of the user with the interface to the DSS, and to consider the impact of the broader task context on performance (Amalberti 1999; Miller, Pelican, and Goldman 2000; Parasuraman 2000; Rasmussen, Pejtersen, and Goldstein 1994). This latter emphasis suggests that the design of a DSS needs to be viewed from the perspective of cooperative problem solving, where the computer and the person interact with and influence each other (Beach and Connally 2005; Hoc 2000; Jones and Mitchell 1995; Larsen and Hayes 2005; Larson 2010; Mital and Pennathur 2004; Parasuraman, Sheridan, and Wickens 2000; Smith, McCoy, and Layton 1997). From this viewpoint, the underlying functionality is just as important as the surface level representation or interface, as it is the overall interaction that determines ultimate performance.

The chapter also frames the design and use of a DSS as a form of cooperative work between several individuals (Bowers, Salas, and Jentsch 2006; Grabowski and Sanborn 2003; Hutchins 1990; Orasanu and Salas 1991; Rasmussen, Brehner, and Leplat 1991; Hutchins 1995; Olson and Olson 1997; Smith et al. 2000), with the computer as the medium through which they cooperate. This teamwork may involve system users who are physically or temporally distributed. In addition, the discussion further suggests that it is useful to think of the design team as working cooperatively with the users, trying to communicate with them and extend or augment their capabilities, and doing so through the software and other artifacts (such as paper documents) that they have developed. This perspective is useful as a reminder that we need to consider the psychology of the designers as well as the psychology of the users of systems, applying that understanding to help us take advantage of the strengths of both groups as they work cooperatively.

Thus, this chapter approaches the design of DSSs from four perspectives. The first is to emphasize the need to identify effective approaches to support human-centered design process, including the application of use cases, CTAs and work domain analyses as a part of an iterative design process. The second is to provide a discussion of the human factors considerations relevant to the design of a DSS (Helander, Landauer, and Prabhu 1997; Meister and Enderwick 2002; Wickens et al. 2004). The third is to provide very practical lists of questions that need to be asked as part of the design process. The fourth is to use case studies to provide concrete, detailed illustrations of how these considerations can be addressed in specific application contexts.

26.2 APPROACHES TO DESIGN

A number of complementary and overlapping approaches have been articulated to help ensure adequate consideration of user needs, preferences and capabilities during the design process, and to gain insights into possible functionality to incorporate in a DSS. These include the use of participatory design (Bodker, Kensing, and Simonsen 2004; Schuler and Namioka 1993), scenario-based design (Carroll 1995), and the completion of various forms of needs assessments (Tobey 2005; Witkin and Altshuld 1995), CTAs and cognitive walkthroughs (Crandall, Klein, and Hoffman 2006; Jonassen, Tessmer, and Hannum 1999; Kirwan and Ainsworth 1992; Rubin, Jones, and Mitchell 1988; Schragen et al. 2000). It also includes various forms of domain analyses aimed at identifying the critical features and constraints that characterize the domain of interest.

There are many approaches to design, all of which are useful and important ways of obtaining the human perspective. One or many of them may be used over the course of a single design project. Some may be most useful in the early stages of the design process, and others as the design becomes more mature. Chapters 48 and 49 discussed the use of participatory and scenario-based design as approaches to the design process. The focus of this section will therefore be on other approaches including various forms of cognitive task and domain analyses. However, it should be noted that the development of, and results from, a CTA can in general be very useful in helping to support participatory design, as can the development and use of scenarios. As Mack (1995, p. 368) notes:

> Developing and sharing a set of touchstone stories is an important cohesive element in any social system. … Gathering, discussing, and sharing these stories as a group can be an effective means to team building.

All the design approaches and techniques discussed can be applied as a part of a larger overall, iterative human-centered design process, so we will start by briefly describing this larger context.

26.2.1 Iterative Human-Centered Design Processes

26.2.1.1 What Makes a Design Process "Human Centered?"

There are many types of design processes, but not all of them are human centered. A *human-centered* design process is one that involves human feedback at all stages of the design process starting from the identification of the need for a DSS and the gathering of user requirements, all the way though testing and evaluation of the final products. Human-centered design processes take into account human skills, needs and limitations. Perhaps the two defining properties of human-centered design processes are that their central goal is to create a result that people find not only usable but highly effective, and the input of potential users is incorporated from conception until launch of the product.

The students of one of the authors once asked, "If we e-mail the customer once a week to let her know what we are doing, does that make it human-centered?" The answer was, "It's a good start, but it's not enough." First of all, the customer is not always one of the potential users. For example, a manager in the Veteran's Administration may ask for a DSS as part of a telemedicine tool to help medical personnel conduct follow up visits across distance. The manager is technically the customer, but not the user in this case. The patients and care providers will be the users. Second, users need to be more actively involved than simply getting a status report each week. The user does not necessarily need to be a member of the design team, as in participatory design, but it is essential to frequently seek user input, review and testing of the design with real or realistic problems.

The advantages of following a human-centered design process are a greater likelihood that users will accept and use the resulting DSS, and that it will provide better support, with greater effectiveness, fewer errors, and increased safety. One may ask, when should one adopt a human-centered versus a nonhuman-centered design approach? We would argue that since all DSSs must interact with people, all DSS should be developed through a human-centered design process. In fact we would further argue that all products, processes and systems interact with humans in some way, and thus all design processes should be human centered.

26.2.1.2 Needs Analysis

Before designing a DSS, one first needs to assess whether a DSS is really needed, and if so what that DSS should do. For example, suppose a manager from a product design and manufacturing company approaches a designer and says, "Many of the product designs my engineers create result in serious manufacturing problems that result in many scrapped products. I want a DSS that will review product designs and give advice on whether or not they are manufacturable." However, what the customer believes will solve the problem may not always be the best solution. The DSS designer must start the design process by gathering information on the overall context using a variety of techniques, including interviews, focus groups, plant walkthroughs, and ethnographic studies of people doing their work. In the situation described above the appropriate approach may be to first interview a variety of engineers (the intended users), and manufacturers to see what they perceive the problem and solution to be, as well as to visit the offices of the designers and manufacturers, and where possible to measure the scope of the problem. How many parts must be scrapped? What were the reasons they had to be scrapped? What is the cost? One may find that a DSS is exactly what is needed or, alternatively that a DSS would be less cost-effective than increasing the training for the engineers on the rapidly evolving manufacturing processes. It is always important to first get a handle on the larger context so as to better understand the underlying problem and goals, and whether a DSS is really the best way to solve the problem.

If the DSS designer finds that a DSS is, indeed, an appropriate solution, then he or she must identify the likely user groups and observe them in their current work environment where possible, and understand the tasks they must do. As a second example, suppose the goal is to design a better scheduling system for receptionists in a medical clinic. To inform design of the DSS, it is useful to sit in an unobtrusive corner of the office as the receptionists are doing their normal work and observe what they do, and where the current problems lie. In one scenario observed by one of the authors, the receptionists had to do multiple conflicting tasks. They had to schedule patients for follow-up appointments at the request of the doctors. These requests were made through the computer. They had to discuss scheduling needs with care providers who would stop into their office. They had to answer telephones, address patient questions, schedule telephone requests for appointments from patients, determine who needed emergency treatment and who could wait based on self-reported pain levels, check in patients as they arrived, and sooth the disgruntled ones. They were often interrupted in the middle of interruptions of other tasks, placing great demands on their short-term memory. Some tasks simply did not get done. The receptionists were extremely stressed, and turnover for the job was high. While it was clear that more was needed than a DSS to really address the scope of the challenge, it was also clear that much could be done to improve the situation through a DSS that helped them to manage multiple scheduling tasks by providing visual interfaces that reduced the burden on their working memories, and assisted with management of simultaneous tasks.

Information from such observations provides critical guidance for the design process, as well as the work domain and CTAs that will be described in Sections 26.2.3 and 26.2.2.1.

26.2.1.3 Design as a Process of Iterative Improvement

Human-centered design processes are typically iterative, with multiple design cycles in which some version of the DSS is created, and potential users are then asked to comment on or test a version of the DSS. Alternatively, a simulation of human behavior can sometimes be used to test or estimate time required for completing a task using the DSS (see Sections 26.2.2.1 and 26.2.2.2). However, since human behavior is very complex, unpredictable and therefore hard to model, in many cases it is simplest and most effective to use human testers.

With each successive design cycle, the prototype of the DSS should become more detailed and realistic, and the testing done with people closer to the real users, with problems in the real contexts in which they will be used. Early versions of the DSS design may be little more than a graph on paper showing the steps in the process that a user would take using the DSS (see the discussions of Sections 26.2.3 and 26.2.2.1). Assessment may involve asking a potential user to review the diagrams by "walking through" the steps for a specific problem and offering feedback (see Section 26.2.2.3). As the design progresses through successive cycles, the design should become progressively more detailed. Later designs might include sketches on paper showing how the visual interface of the software will look, and yet more mature designs may be partially working software prototypes of the DSS. In the final version of the DSS, it should be tested with real users, solving real problems, in the real context. That said, it is often hard to get access to users with special expertise or to recreate the real contexts safely, for example, if the DSS is to be used by military commanders on the battlefield, or by rescue personnel during hurricane relief. However, much can be learned by testing the DSSs with people who have related expertise, such as officers in training, or off-duty or retired coast guard personnel, in situations where people role play to create a "mock" battle or disaster scenario.

There are many prototyping and evaluation techniques which one can apply at various iterations of the design process described throughout this volume. We will focus on cognitive task and work domain analyses as approaches for understanding and modeling the users' work domain, and the cognitive tasks they perform in that domain. These models are critical for informing a DSS design.

26.2.2 Cognitive Task Analyses and Cognitive Walkthroughs

Cognitive task analyses and cognitive walkthroughs are methods for understanding how a task is performed in a specific environment, and assessing DSS designs, sometimes before they are even implemented. A CTA models the components of a task and the thought processes used to carry it out. In some cases, computational versions of cognitive task models, called *cognitive architectures*, are used to predict task performance for a given DSS design. In contrast, a *cognitive walkthrough* uses a potential user, rather than a computational model, to "walk through" the steps in a task as they would be performed using the DSS design. Each provides a different type of useful feedback to the DSS designer. A cognitive architecture may be used to estimate the time required to perform a task given for a proposed DSS. The person performing the cognitive walkthrough may provide more qualitative feedback about whether the design makes sense, and whether there are other functions which the DSS should provide to properly support the task.

26.2.2.1 Cognitive Task Analyses

Scenarios and use cases are also important in providing the framing for specific CTAs. The starting point for a specific

CTA is the high-level context provided by a scenario or use case. As Kutti (1995, p. 19) notes:

> An important feature of a scenario is that it depicts activities in a full context, describing the social settings, resources, and goals of users.

Thus, to begin a CTA, one must specify the person for whom the product is to be designed (a persona), the goal which that person is trying to achieve, and the immediate and broader contexts in which that goal will be pursued.

Many methods for conducting CTAs (Gordon and Gill 1997; Hollnagel 2003) have their roots in hierarchical task analysis techniques, which involve decomposing a high-level goal or task into a hierarchy of subgoals or subtasks (Annett et al. 1971; Annett and Stanton 2000; Diaper and Stanton 2004; Miller, Galanter, and Pribram 1960; Preece et al. 1994). This approach originally focused only on the decomposition of the task without concern with the underlying cognitive processes. However, hierarchical task analysis has been extended to capture not only goal-subgoal relationships, but that also associated cognitive processes.

An influential example of this evolution is GOMS (Goals, Operators, Methods and Selection rules) (Card, Moran, and Newell 1983; Kieras 2004; St. Amant et al. 2005; Williams 2005). GOMS makes use of hierarchical decomposition, but also provides a control structure that introduces operators and rules for selecting which operator to apply in a given goal-driven context. As Preece et al. (1994, pp. 419–420) note:

> A GOMS analysis of human–system interaction can be applied at various levels of abstraction in much the same way that the hierarchical task analysis splits tasks into subtasks. … Three broad levels of granularity determine the GOMS family of models:
>
> - The GOMS model, which describes the general methods for accomplishing a set of tasks.
> - The unit task level, which breaks users' tasks into unit tasks, and then estimates the time that it takes for the user to perform these.
> - The keystroke level, which describes and predicts the time it takes to perform a task by specifying the keystrokes needed.

26.2.2.2 Cognitive Architectures

This move toward explicit computational representation of internal cognitive processes has been extended to the extent that broad, task-independent cognitive architectures have been developed that can be used to model performance in a given task/system context and generate performance predictions (Byrne and Kirlik 2005). Three such systems are Adaptive Control of Thought—Rational (ACT-R) (Anderson 1993; Anderson et al. 2004), State, Operator and Result (SOAR) (Laird, Newell, and Rosenbloom 1987; Nason, Laird, and Schunn 2005), and Executive-Process Interactive Control (EPIC) (Kieras and Meyer 1997; St. Amant et al. 2005). All three incorporate procedural knowledge that must operate within the constraints of certain constructs such as working memory (Baddeley 1998, 1999; Cary and Carlson 2001; Radvansky 2005; Seamon 2005) that are used to model human capabilities and limitations (Myers 2004; Wickens et al. 2004).

26.2.2.3 Cognitive Walkthroughs

This work on the development of computational models to describe or predict task performance is complemented by more qualitative, judgment-based approaches using expert reviews that are similarly rooted in the hierarchical decomposition of a task based on goal-subgoal relationships. These latter techniques are motivated by the assumption that expert judgments about the adequacy of a design will be more accurate and complete if they are made while explicitly considering the context that the user will be dealing with (the user's current goal and subgoal and the display the user is viewing while trying to achieve that goal).

There are a number of variations on how to conduct such qualitative cognitive walkthroughs or predictive CTAs (Annett and Stanton 2000; Bisantz, Roth, and Brickman 2003; Diaper and Stanton 2004; Gordon and Gill 1997; Hollnagel 2003; Jonasson et al. 1999; Kirwan and Ainsworth 1992; Klein 2000; Schraagen, Chipman, and Shalin 2000; Shepherd 2000; Vicente 1999). As an example, one such method is outlined by Lewis and Wharton (1997) as follows:

Step 1. Select the context for a use case to be used for an evaluation. Each such use case specifies three primary considerations:

- *User*. What are the characteristics of the user of the product that might affect its use? (the persona in the use case)
- *Context*. What is the broader physical, organizational, social and legal context in which the product will be used?
- *Task*. What is the high-level task or goal for which the product is being used?

Note that this specification is solution independent. It essentially defines the design problem for this use case.

Step 2. Specify the normative (correct) paths for completing the goal of this use case (represented as a goal hierarchy) using the current product design, indicating the alternative sequences of steps that the user could take to *successfully* achieve the specified goal. Note that there could be more than one correct path for completing a given task.

Step 3. Identify the state of the product and the associated "world" at each node in the goal hierarchy. In the case of a software product, the state of the product would be the current appearance of the interface and any associated internal states (such as the queue of recently completed actions that would be used should an undo function be applied). The state of the "world" applies if the product or the user actually changes something in the world as part of an action, such as changing the temperature of a glass manufacturing system through a process control interface.

Step 4. Generate predictions. For each correct action (node in hierarchy) attempt to

- Predict all the relevant success stories
- Predict all the relevant failure stories
- Record the reasons and assumptions made in generating these stories
- Identify potential fixes to avoid or assist in recovery from failure stories

This final step involves walking through the hierarchy along each path that leads to success, and playing psychologist, generating predicted behaviors at each node (for each associated screen display or product state). This walkthrough could be done by an expert in human–computer interaction (HCI) or by a domain expert. Because of the differences in their backgrounds relevant to the design and the domain of interest, individuals from each group are likely to generate somewhat different predictions.

Lewis and Wharton (1997) suggest asking four questions to help guide this prediction process:

- Will the user be trying to achieve the right effect?
- Will the user notice that the correct action is available?
- Will the user associate the correct action with the desired effect?
- If the correct action is performed, will the user see that progress is being made?

Smith, Stone, and Spencer (2006) suggest some additional, more detailed questions that can be asked to help the analyst consider how different cognitive processes influence performance (Card, Moran, and Newell 1983; Wickens et al. 2004):

- Selective attention: What are the determinants of attention? What is most salient in the display? Where will the user's focus of attention be drawn? (Bennett and Flach 1992; Eckstein 2004; Johnston and Dark 1986; Lowe 2003; Pashler, Johnston, and Ruthruff 2001; Yantis 1998.)
- Automaticity and controlled attentional processes: Does the design support skill-based performance for routine scenarios and controlled, knowledge-based processes for novel situations? (Logan 2005; Rasmussen 1983; Rasmussen, Pejtersen, and Goldstein 1994.)
- Perception: How will perceptual processes influence the user's interpretation? How, for instance, will the proximity of various items on the screen influence judgments of "relatedness" as predicted by the Gestalt Law of Proximity? (Bennett and Flach 1992; Kohler 1992; Tufte 1983, 1990, 1997; Vicente 2002; Watzman 2003.)
- Learning and memory: How will the user's prior knowledge influence selective attention and interpretation? Does the knowledge necessary to perform tasks reside in the world or in the user's mind? (Baddeley 1998, 1999; Grondin 2005; Hester and Garavan 2005; Hutchins 1995; Norman 2002; Ormrod 2003; Radvansky 2005; Scaife and Rogers 1996; Seamon 2005.)
- Information processing, mental models, and situation awareness: What inferences/assumptions will the user make? What internal representations will support such processing? (Endsley 2003; Endsley and Garland 2000; Gentner and Stevens 1983; Johnson et al. 1981; Plous 1993; Schaeken 2005.)
- Design-induced error: How could the product design and the context of use influence performance and induce errors? (Bainbridge 1983; Beach and Connally 2005; Johnson et al. 1981; Larsen and Hayes 2005; Parasuraman and Riley 1997; Reason 1990, 1997; Roth, Bennett, and Woods 1987; Sheridan 2002; Skirka et al. 1999; Smith and Geddes 2003; Smith et al. 1997.)
- Motor performance: Can the controls be used efficiently and without error? (Jagacinski and Flach 2003.)
- Group dynamics: How will the system influence patterns of interaction among people? (Baecker et al. 1995; Brehm, Kassin, and Fein 1999; Forsyth 1998; Levi 2001; Olson and Olson 2003.)

Thus, Steps 1–3 as described earlier involve describing how a product (such as a DSS) can be used to achieve a specific goal. Step 4 involves predicting how people might perform when using this product, either successfully or unsuccessfully, as they try to achieve this goal. Thus, the "cognitive" part of this analysis is embedded only in Step 4. Note also that the goal-subgoal hierarchy does not attempt to represent the further sequences of actions that could arise once the user starts down a failure path. This helps to keep the size of the hierarchy smaller, keeping the analysis task more manageable. (For some applications, it may be desirable to actually expand the goal-subgoal hierarchy to represent the complete path of a failure story.)

Finally, going beyond just the use of hierarchical decomposition techniques, Gordon and Gill (1997, pp. 133–134) note that

> CTA [Cognitive Task Analysis] methodologies differ along several dimensions. These include the type of knowledge *representation* or formalism used to encode the information obtained from the analysis, the *methods* used for eliciting or identifying expert (or novice) knowledge, and the *type of task and materials* used in the analysis. Typical representation formats include lists and outlines, matrices and cross-tabulation tables, networks, flowcharts, and problem spaces with alternative correct solution paths.

Thus, although the use of hierarchical goal-subgoal representations is one of the more popular approaches, there are a variety of other complementary approaches that are labeled as CTAs (Crandall, Klein, and Hoffman 2006). These alternatives include the use of concept maps and conceptual graph analyses (Gordon and Gill 1992) and different approaches to extracting and representing the knowledge of experts about critical incidents (Crandall, Klein, and Hoffman 2006; Klein, Caulderwood, and MacGregor 1989). The ultimate goal of all of these approaches, however, is to better represent and predict how users' knowledge and control processes or cognitive processes will be applied in a given task environment. This task environment could be the existing world (a descriptive analysis) that is being studied in order to design improved support tools, or an envisioned world that incorporates some specific design proposal intended to improve performance (a predictive analysis).

26.2.3 Work Domain Analyses

Cognitive task analyses tend to focus on how to support performance for an already identified task or goal. However, in many cases DSSs are designed to support performance in very complex environments (Bar-Yam 2003, 2005) where it is unlikely that all of the possible scenarios that could arise will be predicted during the design process. Thus, there is also a need for design methodologies that support the development of robust designs that will be effective even when novel situations arise (Vicente 1999).

To deal with this need, a complementary set of design methodologies have been developed under the label of work domain analysis (Burns, Bizantz, and Roth 2004; Burns and Hajdukiewicz 2004; Burns and Vicente 2001; Reising and Sanderson 2002; Vicente 2002; Vicente and Rasmussen 1990; Vicente and Rasmussen 1992). In general terms, the emphasis is on understanding and modeling the work domain or application area rather than simply modeling the performance of people on a prespecified set of tasks within this domain. The argument is that the domain analysis can identify the critical parameters or constraints relevant to successful performance in that domain and thus serve as the basis for a design that can handle both routine and novel (unanticipated) situations.

To make this point more concrete, as a very simple but illustrative example, Vicente (2000) contrasts verbal directions on how to travel from one point to another versus the use of a map. In this analogy, the directions provide a very efficient but brittle description of how to accomplish the desired goal. The map requires more work (interpretation), but is more flexible in the face of obstacles that might render the directions unusable. Building on this analogy, Vicente (2000, p. 115) notes:

> This decrease in efficiency [of maps vs. directions] is compensated for by an increase in both flexibility and generality. Like maps, work domain representations are more flexible because they provide workers with the information they need to generate an appropriate response online in real time to events that have not been anticipated by designers. This is particularly useful when an unforeseen event occurs because task representations, by definition, cannot cope with the unanticipated.

The literature on work domain analyses discusses the use of abstraction (means-end) hierarchies and aggregation (part-whole) hierarchies to complete a domain analysis, and then using the results of this analysis to guide the design of the DSS for supporting operator performance (Burns and Hajdukiewicz 2004; Vicente 1999; Vicente and Rasmussen 1990). The basic logic of this approach is that, in order to design appropriate representations to assist an operator in completing tasks using some system, the complete and correct semantics of the domain must first be appropriately defined. Since such a design is concerned with supporting goal-oriented behaviors, one way to represent the semantics is with an abstraction (means-end) hierarchy that captures the "goal-relevant properties of the work domain" (Vicente and Rasmussen 1990, p. 210). In addition, because the domains of interest are often quite complex, aggregation hierarchies may be useful to decompose the overall domain into a set of interrelated subparts.

26.2.4 Approaches to Design—Summary

Much of the work on methods for completing CTAs and work domain analyses has been stimulated by the challenges of developing DSSs to support performance on complex tasks. The description above provides a sense of the range of techniques that have been developed, and emphasizes the need to understand both the constraints of the application context and the nature of the agent(s) (person or people) involved. The insights provided by such analyses can be used to guide the design of the underlying functionality embedded within a DSS, and to guide the design of the interaction between a DSS and its user.

26.3 HUMAN PERFORMANCE ON DECISION-MAKING TASKS

There are several reasons why an understanding of human performance is important to the designer of a DSS. The first is that the motivation for developing such systems is to increase efficiency (reduce production costs or time) and/or to improve the quality of performance. Thus, it is important for the design team to be able to efficiently and effectively complete the initial problem definition and knowledge engineering stages of a project, identifying areas where improvements are needed. An understanding of human performance makes it possible to, in part take a top-down approach to this, looking to see whether certain classic human performance phenomena (such as hypothesis fixation) are influencing outcomes in a particular application. A second motivation for understanding human performance is that a human-centered approach to the design of a DSS requires consideration and support of the user's skills (Garb 2005). To do this effectively, knowledge of human performance (perception, learning and memory, problem solving, decision making, etc.) is essential. A number of these aspects of human performance are covered in Chapters 1–5, so this chapter will just highlight some of the factors most relevant to the design of a DSS, and discuss how such factors are relevant to this design task.

26.3.1 Errors and Cognitive Biases

In very broad terms, human errors (Strauch 2004; Wiegmann and Shapell 2003) can be classified as slips and as mistakes (Norman 1981). Slips arise through a variety of cognitive processes, but are defined as behaviors where the person's actions do not match his intentions. Generally, this refers to cases where the person has the correct knowledge to achieve some goal but, due to some underlying perceptual, cognitive or motor process, fails to correctly apply this knowledge. As Norman (1988, p. 106) describes it: "Form an appropriate goal but mess up in the performance, and you've made a slip." Mistakes, on the other hand, refer to errors resulting from the accurate application of a person's knowledge to achieve some goal, but where that knowledge is incomplete or wrong.

DSSs are potentially useful for dealing with either of these sources of errors. If slips or mistakes can be predicted by the design team, then tools can be developed to either help prevent them, recover from them, or reduce their impacts.

26.3.1.1 Slips

Norman (2002) discusses six categories of slips. Knowing something about these different causes can help the designer to look for possible manifestations in a particular application area. These six categories as defined in Norman include the following:

- Capture errors, "in which a frequently done activity suddenly takes charge instead of (captures) the one intended" (p. 107).
- Description errors, where "the intended action has much in common with others that are possible" and "the internal description of the intention was not sufficient" often resulting in "performing the correct action on the wrong object" (pp. 107–108).
- Data-driven errors, where an automatic response is triggered by some external stimulus that triggers the behavior at an inappropriate time.
- Associative activation errors where, similar to a data-driven error, something triggers a behavior at an inappropriate time, but in this case the trigger is some internal thought or process.
- Loss-of-activation errors, or "forgetting to do something" (p. 109).
- Mode errors, or performing an action that would have been appropriate for one mode of operation for a system, but is inappropriate for the actual mode or state that the system is in.

26.3.1.2 Mistakes

As defined earlier, mistakes are due to incorrect knowledge (the rule, fact, or procedure that the person believes to be true is incorrect, resulting in an error) or incomplete (missing) knowledge.

26.3.1.3 Cognitive Biases

The literature on human error also provides other useful ways to classify errors in terms of surface level behavior or the underlying cognitive process. Many of these are discussed under the label of cognitive biases (Brachman and Levesque 2004; Bradfield and Wells 2005; Fraser, Smith, and Smith 1992; Gilovich, Griffin, and Kahneman 2002; Haselton, Nettle, and Andrews 2005; Kahneman and Tversky 2000; Plous 1993; Poulton 1989), including

- Gambler's fallacy (Croson and Sundali 2005). Concluding there is a pattern in a series of events that is in reality a random sequence. Gilovich, Vallone, and Tversky (1985) for instance, found that individuals believe they see streaks in basketball shooting even when the data show that the sequences are essentially random.
- Insensitivity to sample size (Bjork 1999). Failing to understand that the law of large numbers implies that the probability of observing an extreme result in an average decreases as the size of the sample increases. Tversky and Kahneman (1974), for example, found that subjects believed that "the probability of obtaining an average height greater than 6 feet was assigned the same value for samples of 1000, 100 and 10 men."
- Incorrect revision of probabilities. Failure to revise probabilities sufficiently when data are processed simultaneously, or conversely, revising probabilities too much when processing the data sequentially.
- Ignoring base rates. Failure to adequately consider prior probabilities when revising beliefs based on new data.
- Use of the availability heuristic (Blount and Larrick 2000). Tversky (1982) suggests that the probability of some type of event is in part judged based on the ability of the person to recall events of that type from memory, thus suggesting that factors like recency may incorrectly influence judgments of probability.
- Attribution errors. Jones and Nisbett (1971) describe this by noting that "there is a pervasive tendency for actors to attribute their actions to situational requirements, whereas observers tend to attribute the same actions to stable personal dispositions."
- Memory distortions due to the reconstructive nature of memory (Blank 1998; Handberg 1995). Loftus (1975) describes processes that distort memories based on the activation of a schema as part of the perception of an event, and the use of that schema to reconstruct the memory of the event based on what the schema indicates should have happened rather than what was actually perceived originally. Smith et al. (1986) and Pennington and Hastie (1992) provide descriptions of similar phenomena in decision-making tasks.

In recent years, there has been considerable controversy regarding the nature and validity of many of these explanations as "cognitive biases" (Fraser, Smith, and Smith 1992; Koehler 1996; Haselton, Nettle, and Andrews 2005), suggesting that it is important to carefully understand the specific context in order to generate predictions as to whether a particular behavior will be exhibited. Using the incorrect revision of probabilities as an illustration, Navon (1978) suggested that in many real-world settings the data are not independent, and that what appears to be conservatism in revising probabilities in a laboratory setting may be the result of the subjects applying a heuristic or cognitive process that is effective in real-world situations where the data are correlated.

A more detailed example is provided by the popular use of the label "hypothesis fixation." This behavior refers to some process that leads the person to form an incorrect hypothesis and to stick with that hypothesis, failing to collect critical data to assess its validity or to revise it in the face of conflicting data. A variety of cognitive processes have been hypothesized to cause this behavior. One example is called biased assimilation, where the person judges a new piece of data as supportive of his hypothesis (increasing the level of confidence in the hypothesis) based simply on the consideration of whether that outcome could have been produced under his hypothesis. This contrasts with inferential processes based on a normative model that suggest that beliefs should be revised based on the relative likelihood of an outcome under the possible competing hypotheses, and that would, in the same circumstance, lead to a reduction in the level of confidence in his hypothesis.

Another example of a cognitive process discussed in the literature as relevant to hypothesis fixation is the so-called "confirmation bias." This phenomenon, described in Wason (1960) and in Mynatt et al. (1978), is concerned with the person's data collection strategy. As an example, in a study asking subjects to discover the rules of particle motion in a simulated world, Mynatt et al. described performance by concluding that there was "almost no indication whatsoever that they intentionally sought disconfirmation." Later studies, however, have suggested that this "bias" is really an adaptive "positive test" strategy that is effective in many real-world settings, and that it is the unusual nature of the task selected by Mynatt et al. that makes it look like an undesirable bias (Klayman and Ha 1987). Smith et al. (1986) further suggest that, in real-world task settings where such a strategy might lead to confirmation of the wrong conclusion, experts often have domain-specific knowledge or rules that help them to avoid this strategy altogether, thus ensuring that they collect critical potentially disconfirming evidence.

26.3.1.4 Designer Error

Many of the error-producing processes discussed above appear to focus on system operators. It is important to recognize, however, that the introduction of a DSS into a system is a form of cooperative work between the users and the design and implementation team and that, like the users, the design team is susceptible to errors (Petroski 1994). These errors may be due to slips or mistakes. In the case of mistakes, it may be due to inadequate knowledge about the application area (the designers may fail to anticipate all of the important scenarios) or due to incorrect knowledge. It may also be due to a failure to adequately understand or predict how the users will actually apply the DSS, or how its introduction will influence their cognitive processes and performances. In addition, sources of errors associated with group dynamics may be introduced. Whether such errors are due to a lack of sufficient coordination, where one team member assumes that another is handling a particular issue, or due to the influence of group processes (Brehm, Kassin, and Fein 1999; Forsyth 1998; Janis 1982; Levi 2001; McCauley 1989; Tetlock et al. 1992; Tetlock 1998) on design decisions, the result can be some inadequacy in the design of the DSS, or in the way the user and the DSS work together. (Note that such group processes are also important potential sources of errors when system operators work together as part of teams, with or without technological support.)

26.3.1.5 Systems Approaches to Error

Reason (1991, 1997) and Wiegmann et al. (2004) caution designers against fixating on the immediately preceding "causes" of an accident when trying to prevent future occurrences. They remind us that, in many system failures, a number of conditions must coincide in order for the failure to occur. This presents a variety of leverage points for preventing future occurrences, many of which are preemptive and thus remote from the actual accident or system failure. Many of these changes focus on preventing the conditions that could precipitate the system failure, rather than improving performance once the hazardous situation has been encountered.

26.3.1.6 Errors and Cognitive Biases—Implications for Design

As described at the beginning of this section, the value of this literature to the designer is the guidance it provides in conducting the initial knowledge engineering studies that need to be completed in order to identify opportunities for improvement in some existing decision-making process that is to be supported by a new DSS. Familiarity with this literature is also critical to the designer to ensure that the introduction of the DSS does not result in new design-induced errors. This literature also serves to emphasize two additional considerations:

- The occurrence of errors are often due to the co-occurrence of a particular problem-solving strategy with a given task environment that is "unfriendly" to that strategy (i.e., situations where the strategy is not sufficiently robust). Strategies that were adaptive in one setting may no longer be adaptive in the newly designed environment. The potential for such a negative transfer of learning needs to be considered as part of the design.

- For routine situations, people tend to develop expertise that lets them employ knowledge rich problem-solving strategies (Crandall, Klein, and Hoffman 2006; Klein 1993; Kolodner 1993; Newell 1990) that avoid the errors that could be introduced by certain general problem-solving strategies. However, in many system designs, people are expected to act as the critical safety net during rare, idiosyncratic events that the design team has failed to anticipate. These are exactly the situations where people have to fall back on their general problem-solving strategies, and are thus susceptible to the potential errors associated with these weak problem-solving methods.

In addition, this review emphasizes the need to take a broad systems perspective, recognizing that the design and implementation team are a potential source of error, and understanding that, if the goal is to prevent or reduce the impact of errors, then the solution is not always to change performance at the immediate point where an error was made. It may be more effective to change other aspects of the system so that the potentially hazardous situation never even arises.

Finally, the emphasis on designer error implies that the design process for a DSS needs to be viewed as iterative and evolutionary. Just because a tool has been fielded, it is not safe to assume that no further changes will be needed. As Horton and Lewis (1991) discuss, this has organizational implications (ensuring adequate communications regarding the actual use of the DSS; budgeting resources to make future revisions), as well as architectural implications (developing a system architecture that enables revisions in a cost-effective manner).

26.3.2 Human Expertise

As discussed in Chapters 2–4, the nature of human information processing imposes a variety of constraints that influence how effectively a person processes certain kinds of information. These include memory, perceptual and information processing constraints. As a result, there are certain decision tasks where the computational complexity or knowledge requirements limit the effectiveness of unaided individual human performance.

From a design perspective, these information processing constraints offer an opportunity. If important aspects of the application are amenable to computational modeling, then a DSS may provide a significant enhancement of performance, either in terms of efficiency or the quality of the solution. (Note that even if development of an adequate computational model is not feasible, there may be other technological improvements, such as more effective communications environments, that could enhance performance. However, this chapter is focusing on DSSs that incorporate active information processing functions by the software.)

A good example of this is flight planning for commercial aircraft. Models of aircraft performance considering payload, winds, distance and aircraft performance characteristics (for the specific aircraft as well as the general type of aircraft) are sufficiently accurate to merit the application of DSSs that use optimization techniques to generate alternative routes and altitude profiles to meet different goals in terms of time and fuel consumption (Smith, McCoy, and Layton 1997). This example also provides a reminder, however, that it is not just human limitations that are relevant to design. It is equally important to consider human strengths, and to design systems that complement and are compatible with users' capabilities. In flight planning, for instance, this includes designing a system that allows the person to incorporate his judgments into the generation and evaluation of alternative flight plans, considering the implications of uncertainty in the weather, air traffic congestion and other factors that may not be incorporated into the model underlying the DSS.

This cooperative systems perspective has several implications. First, the designer needs to have some understanding of when and how the person should be involved in the alternative generation and selection processes. This requires insights into how the person makes decisions, in terms of problem-solving strategies as well as in terms of access to the relevant knowledge and data, and how the introduction of a DSS can influence these problem-solving processes. It also implies that the strengths underlying human perceptual processes need to be considered through display and representation aiding strategies (Burns and Hajdukiewicz 2004; Kleinmuntz and Schkade 1993; Jones and Schkade 1995; Tufte 1997) in order to enhance the person's contributions to the decision-making process by making important relationships more perspicuous or salient.

26.3.2.1 Descriptive Models

The literature on human problem solving and decision making provides some very useful general considerations for modeling human performance within a specific application. This literature, which is reviewed in more detail in Chapters 2–3, covers a variety of descriptive models of problem solving. This includes the use of heuristic search methods (Clancey 1985; Dasgupta, Chakrabarti, and Desarkar 2004; Michalewicz and Fogel 2004; Michie 1986; Rayward-Smith et al. 1996; Russell and Norvig 1995). This modeling approach (Newell and Simon 1972) conceptualizes problem solving as a search through a space of problem states and problem-solving operations to modify and evaluate the new states produced by applying these operations. The crucial insight from modeling problem solving as search is an emphasis on the enormous size of the search space resulting from the application of all possible operations in all possible orders. In complex problems, the size of the space precludes exhaustive search of the possibilities to select an optimal solution. Thus, some heuristic approach that satisfies, such as elimination by aspects (Tversky 1972), is needed to guide a selective search of this space, resulting in the identification of an acceptable (if not optimal) solution.

Another aspect of problem solving that Newell and Simon noted when they formulated the search paradigm was the importance of problem representation. Problem representation issues emphasize that a task environment is not inherently objectively meaningful, but requires interpretation for problem solving to proceed. Some interpretations or representations are more likely to lead to successful solutions than others (for certain agents). In fact, an important component of expertise is the set of features or the representation used to characterize a domain (Lesgold et al. 1988), sometimes referred to as domain ontology.

Task-specific problem spaces allow problem solvers to incorporate task- or environment-induced constraints, focusing attention on just the operations of relevance to preselected goals (Sewell and Geddes 1990). For such task-specific problem solving, domain-specific knowledge (represented in computational models as production rules, frames, or some other knowledge representation) may also be incorporated to increase search efficiency or effectiveness (Dechter 2003; Ghallib, Nau, and Traverso 2004; Laird, Newell, and Rosenbloom 1987; Shalin et al. 1997).

These models based on symbolic reasoning have been specialized in a number of powerful ways for specific generic tasks like diagnosis or planning (Chandrasekaran 1988; Ghallib, Nau, and Traverso 2004; Gordon 2004; Miller, Galanter, and Pribram 1960; Mumford, Schultz, and Van Doorn 2001; Nareyek, Fourer, and Freuder 2005). For example, computational models of planning focus on the use of abstraction hierarchies to improve search efficiency (Sacerdoti 1974), dealing with competing and complementary goals (Wilenski 1983) and mixed top-down/bottom-up processing (Hayes-Roth and Hayes-Roth 1979) to opportunistically take advantage of data as it becomes available. More recent developments have sought to deal with planning in stochastic environments (Madani, Condon, and Hanks 2003; Majercik and Littman 2003).

In contrast to such sequential search models, models based on case-based reasoning, recognition primed decision making or analogical reasoning (Crandall, Klein, and Hoffman 2006; Klein 1993; Kolodner 1993; Riesbeck and Schank 1989) focus on prestructured and indexed solutions based on previously experienced situations. These modeling approaches suggest that human experts in complex operational settings rarely describe their cognitions as sequential searches to construct alternative solution states, but rather as recognition processes that match features of the situation to prototypical, preformulated response plans (Zuboff 1988; Rasmussen 1983). In complex dynamic domains (Oliver and Roos 2005) like aviation or firefighting, these preformulated plans incorporate a great deal of implicit knowledge reflecting the constraints of the equipment, the environment, other participants in the work system, and previous experience. Over time, solutions sensitive to these constraints result in familiar, accepted methods which are then triggered by situational cues and modified in small ways as needed to deal with the specific scenario. However, it is important to recognize that such recognition-based processes can cause familiar task features to invoke only a single interpretation, resulting in hypothesis fixation or hindering creative departure from normal solutions. This limitation can be critical when the task environment contains a new, unexpected feature that requires a new approach.

26.3.2.2 Normative Optimal Models

In addition to these descriptive models, there is a large literature that characterizes human decision making relative to various normative models based on optimal processes, which help to emphasize important factors within the task and task environment that should be considered in making a decision in order to arrive at a better decision. These normative models are based on engineering models such as statistical decision and utility theory, information theory, and control theory (Jagacinski and Flach 2003; Rouse 1980; Sheridan and Ferrell 1974). By contrasting human performance with optimal performance on certain tasks and emphasizing the factors that should influence decision making in order to achieve a high level of performance, this literature helps the designer to look for areas where human performance may benefit from some type of DSS.

Finally, the literature on human expertise (Charness et al. 2006) emphasizes the ability of people to learn and adapt to novel situations, and to develop skeletal plans that guide initial decisions or plans, but that are open to adaptation as a situation unfolds (Geddes 1989; Suchman 1987; Wilenski 1983). The literature also emphasizes variability, both in terms of individual differences and in terms of the ability of a single individual to use a variety of decision-making strategies in some hybrid fashion in order to be responsive to the idiosyncratic features of a particular scenario.

26.3.2.3 Human Expertise—Implications for Design

An understanding of the literature on human expertise is of value to the designer in a number of different ways (Chi et al. 1998). First, in terms of the initial problem-definition and knowledge engineering stages, familiarity with models of human problem solving and decision making can help guide the designer in looking for important features that influence performance within that application. Second, in an effort to provide cognitive compatibility with the users, the design of many of the technologies underlying DSSs is guided by these same computational models of human performance. Third, even if the underlying technology is not in some sense similar to the methods used by human experts in the application, the designer needs to consider how the functioning of the DSS system should be integrated within the user's decision-making processes.

In terms of some specific emphases, the above discussion suggests the following:

- Do not fixate on active DSS technologies as the only way to improve system performance. Enhancing human performance through changes in procedures, improvements in passive communication tools, better external memory aids, and so on, may be more cost-effective in some cases. In addition, such changes may be needed to complement an active DSS in order to make its use more effective.

- Look for ways in which the direct perception of ecological constraints allows people to perform expertly (Flach et al. 1995). The ability to perceive these critical parameters or constraints directly may make what seems like a very difficult information processing task much less demanding.
- Related to the consideration of ecological constraints was the earlier suggestion that problem representation has a strong influence on how easily a problem solver can find a good solution. This issue of perspicuity (salience of the problem solution), however, is dependent not only on the characteristics of the task and task environment, but also on the nature of the problem solver. Thus, in order to enhance human performance as part of the decision-making process, it is important to consider alternative ways of representing the situation that will enable human perceptual and cognitive processes to work more effectively. In addition, problem representation can affect the performance of the designer. Consequently, alternative problem representations need to be generated as part of the design process to help the designer think more effectively.
- Consider the applicability of normative optimal models of performance for the task in order to focus attention on factors that should be influencing current performance, and to contrast optimal strategies with the strategies that people are actually using in response to these task-determined factors. Aspects of these normative models may also be appropriate for more direct inclusion in the DSS itself, even though the strategies used by people do not fully reflect the optimal processes highlighted by these normative models. In addition to considering the task structure highlighted by normative optimal models, use knowledge of the variety of different descriptive models of human problem solving and decision making to guide knowledge engineering efforts, making it easier and more efficient to understand how people are currently performing the tasks. Thus, all of these models of decision making represent conceptual tools that help the designer to more effectively and efficiently understand performance in the existing system, and to develop new tools and procedures to improve performance.
- Whether the DSS is designed to process information "like" the user at some level, or whether it uses some very different processing strategy, the user needs to have an appropriate and effective mental model of what the system is and is not doing (Kotovsky, Hayes, and Simon 1985; Lehner and Zirk 1987; Nickerson 1988; Zhang 1997; Zhang and Norman 1994). To help ensure such cognitive compatibility, the designer therefore needs to understand how people are performing the task.
- Consider design as a prediction task, trying to predict how users will interact with the DSS system, and how it will influence their cognitive processes and performances.

26.3.3 Additional Cognitive Engineering Considerations

Sections 26.3.1–26.3.2 focused on our understanding of human performance on decision-making tasks and the implications of that literature for design. Below, some additional considerations based on studies of the use of DSSs are presented.

26.3.3.1 Human Operator as Monitor

Numerous studies make it clear that, given the designs of DSSs for complex tasks must be assumed to be brittle, there is a problem with designs that assume that human expertise can be incorporated as a safety net by asking a person to simply monitor the output of the DSS, with the responsibility for overriding the software if a problem is detected. One problem with this role is that, in terms of maintaining a high level of attentiveness, people do not perform well on such sustained attention tasks (Meister and Enderwick 2002). As Bainbridge (1983, p. 776–7) notes: "We know from many 'vigilance' studies (Mackworth 1950) that it is impossible for even a highly motivated human being to maintain effective visual attention toward a source of information on which very little happens, for more than about half an hour. This means that it is humanly impossible to carry out the basic function of monitoring for unlikely abnormalities."

A related issue is the problem of loss of skill. As Bainbridge further notes, if the operator has been assigned a passive monitoring role, he "will not be able to take over if he has not been reviewing his relevant knowledge, or practicing a crucial manual skill." Thus, a major challenge for retaining human expertise within the system is "how to maintain the effectiveness of the human operator by supporting his skills and motivation."

26.3.3.2 Complacency and Overreliance

Studies reported by Parasuraman and Riley (1997) and Parasuraman (2000) introduce further concerns about assigning the person the role of critiquing the computer's recommendations before acting. These studies discuss how the introduction of a DSS system can lead to overreliance by the human user when the software is generating the initial recommendations (Metzger and Parasuraman 2005; Skitka, Mosier, and Burdick 1999).

26.3.3.3 Excessive Mental Workload

Designs that relegate the person to the role of passive monitor run the risk of a vigilance decrement due to insufficient engagement and mental workload. At the other extreme, designers must recognize that "clumsy automation" that leaves the person with responsibility for difficult parts of the task (such as coping with an emergency), but that adds additional workload (Kushleyeva et al. 2005) due to the awkward interactions now required to access information and functions embedded in the DSS (such as navigating through complex menus to view certain information), can actually impair performance because of the added mental workload of interacting with the DSS (Wiener and Nagel 1988). One line of research that is now getting increased attention is the potential to design adaptive systems that can monitor the human operator using neurophysiological measures that attempt to detect problems with alertness, mental workload or attention (Caggiano and Parasuraman 2004; Freeman, Mikulka, and Scerbo 2004; Grier, Warm, and Dember 2003; Mikulka, Scerbo, and Freeman 2002; Luo, Greenwood, and Parasuraman 2001). This work seeks to "determine whether a biocybernetic, adaptive system could enhance vigilance performance," with the goal of improving "monitoring performance on critical activities such as air traffic control and radar and sonar operation" (Mikulka, Scerbo, and Freeman 2002, p. 654), or offloading work to software or other people when the computer detects that excessive attentional demands are being placed on an individual.

26.3.3.4 Lack of Awareness or Understanding

Even if the person is sufficiently engaged with the DSS and the underlying task, designers need to consider how to ensure that the user has an accurate mental model of the situation, and of the functioning and state of the DSS (Billings 1997; Larkin and Simon 1987; Mitchell and Miller 1986; Roth, Bennett, and Woods 1987; Sarter and Woods 1993). If the person does not have such an understanding, then it may be difficult for him to intervene at appropriate times or to integrate the computer's inputs into his own thinking appropriately (Geddes 1997b). This concern has implications for selection of the underlying technology and conceptual model for a system, as well as for the design of the visual or verbal displays intended to represent the state of the world and the state of the software for the user, including explanations of how the DSS has arrived at its recommendations (Clancey 1983; Hasling, Clancey, and Rennels 1984; Nakatsu and Benbasat 2003).

26.3.3.5 Lack of Trust and User Acceptance

As outlined earlier, overreliance can be a problem with certain assignments of roles to the person and the computer. At the other extreme, lack of trust or acceptance of the technology can eliminate or reduce its value (Clarke et al. 2006; Muir 1987). This lack of acceptance can result in outright rejection of the DSS (in which case it either is not purchased or not used), or in a tendency to underutilize it for certain functions. It is important to note that this lack of acceptance can be due to resistance to change (Cartwright and Zander 1960; Forsyth 1998; Levi 2001) even if there is no intrinsic weakness in the DSS, due to general beliefs held by the operators (rightly or wrongly) about how the software will influence their lives, or due to beliefs about how well such a DSS can be expected to perform (Andes and Rouse 1992; Lee and See 2004; Marsh and Dibben 2003; Muir and Moray 1996; Riegelsberger, Sasse, and McCarthy 2005).

26.3.3.6 Active Biasing of the User's Cognitive Processes

Complacency is one way in which a DSS can influence the person's cognitive processing. Studies have also shown, however, that the use of a DSS can also actively alter the user's cognitive processes (Beach and Connally 2005; Larsen and Hayes 2005; Smith, McCoy, and Layton 1997). The displays and recommendations presented by the software have the potential to induce powerful cognitive biases, including biased situation assessment and failures by the user to activate and apply his expertise because normally available cues in the environment are no longer present. The net result is that the person fails to exhibit the expertise that he would normally contribute to the decision-making task if working independently, not because he lacks that expertise, but because the computer has influenced his cognitive processes in such a way that this knowledge is never appropriately activated. Studies have shown that these biasing effects can induce practitioners to be 31% more likely to arrive at an incorrect diagnosis on a medical decision-making task (Guerlain et al. 1996) and 31% more likely to select a very poor plan on a flight planning task (Smith, McCoy, and Layton 1997).

26.3.3.7 Distributed Work and Alternative Roles

Much of the literature focuses on the interactions between a single user and the DSS. Increasingly, however, researchers and system developers are recognizing that one approach to effective performance enhancement is to think in terms of a distributed work paradigm (Schroeder and Axelsson 2005; Smith, McCoy, and Orasanu 2001), where the software may be one "agent" in this distributed system (Geddes and Lizza 1999), or where the software may be viewed primarily as a mediator to support human–human interactions within virtual or distributed teams (Baecker et al. 1995; Bowers, Salas, and Jentsch 2006; Caldwell 2005; Carroll et al. 2003; Handy 1995; Hertel, Geister, and Konradt 2005; Hinds and Kiesler 2002; Hinds and Mortensen 2005; Hutchins 1990, 1995; Katz et al. 2004; Lurey and Raisinghani 2001; Olson and Olson 1997, 2003; Orasanu and Salas 1993; Rasmussen, Brehner, and Leplat 1991; Salas, Bowers, and Edens 2001; Salas and Fiore 2004; Smith et al. 1997). Such distributed systems can now be found in the military, in medicine, in aviation and a variety of other application areas.

Sycara and Lewis (2004) suggest that

> There are three possible functions that software agents might have within human teams:
>
> 1. Support the individual team members in completion of their own tasks
> 2. Assume the role of a (more or less) equal team member by performing the reasoning and tasks of a human teammate
> 3. Support the team as a whole (p. 204), and that "teams could be aided along four general dimensions: accessing information, communicating, monitoring, and planning" (p. 227).

In considering the design and use of DSSs for such applications, all of the more traditional factors must be considered. In addition, questions regarding the impact of the technology on team situation awareness (Endsley 2003), rapport, trust, and communication become paramount. It is also important to think of a broader set of design parameters, as distributed approaches to work open up the potential to use a variety of different architectures for distributing the work in terms of the locus of control or responsibility and the distribution of knowledge, data, and information processing capabilities (Griffith, Sawyer, and Neale 2003; Sheridan 1997; Sheridan 2002; Smith et al. 1997; Smith McCoy, and Orasanu 2000; Smith et al. 2003). Furthermore, DSSs may be necessary to make effective certain architectures for distributing the work.

26.3.3.8 Organizational Failures

The discussions above focus on design-induced errors made by the system operators. It is equally important for the design team to recognize that part of the design process is ensuring that the management of the organization into which the DSS is introduced will provide an effective safety net. This means that the design team needs to ensure that the organization has established effective procedures to detect significant problems associated with the introduction and use of the DSS, and that such problems are communicated to the levels of management where responsibility can and will be taken to respond effectively to these problems (Horton and Lewis 1991).

26.3.3.9 Social Responses to Computers

An accumulating body of research indicates that people respond to computers with strong social responses even when they consciously claim to think of computers as machines (Nass, Steuer, and Tauber 1994; Wang et al. 2005; Hayes and Miller 2010). For example, Johnson found that when the computer tutor, Adele, corrected learners with the same wording multiple times, they commented that Adele had a stern personality and low regard for their work (Johnson et al. 2003). As a second example, when computer agents are given human voices or faces, users often apply the same prejudices to the computer agents which they apply to different races or genders (Nass, Moon, and Green 1997; Gong 2008).

However, people have social responses to computers, regardless of whether the computers have anthropomorphic forms, faces, or voices (Nass, Steuer, and Tauber 1994), although the effect may be stronger when they are anthropomorphic. It is important for DSS designers to understand this, because users frequently react to computer applications as if they were social beings rather than simply "dumb" machines. Conversely, they may expect computer applications to interact with them according to culturally accepted norms for politeness (Hayes and Miller 2010). Software that does not follow these norms, for example by interrupting at inappropriate times, often irritates users and may lead to disuse of that application. Consider "help" applications that pop-up on the screen on their own, interrupting one's flow of thought, and requiring action to get rid of them. Such "helpful" programs are often turned off. However, following these norms may increase users' willingness to accept and use an application while increasing that application's effectiveness. For example, students may learn more from Adele if they feel respected and supported by her; users may be more motivated to continue and maintain exercise programs if their computer exercise advisor follows conventions for conversations (opening with small talk), and shows appropriate expressions (Bickmore 2010). The challenge in programming DSSs and other computer applications to follow culturally accepted norms for interaction is that many of these norms are implicit. We use them often without consciously being aware of what they are.

There are numerous applications where the utility of a DSS is enhanced if the user develops a certain appropriate level of trust in the DSS, and/or a long-term "relationship" with it. For example, "smart" medicine dispensers that remind patient's when they have skipped a dose (Wu et al. 2010); medical robots which monitor patient health, bring meals or medication, or assist patients in walking (Zhang, Zhu, and Kaber 2010); message managers used by dismounted soldiers (Dorneich et al. 2010); computer tutors (Johnson and Wang 2010); and exercise coaches whose success depends on their ability to engage the patient and keep them motivated over periods of months or years to continue an exercise program (Bickmore 2010). Norms for social interaction in a computer agent with a face, such as a computer tutor, health coach, or medical robot may extend to more than just what they do and say, but also to facial expressions. For example, the computer agent should not smile or laugh when the patient is explaining his or her frustration with his or her inability to manage pain.

In such applications, it has been demonstrated experimentally that a DSS can help soldiers to manage critical information more effectively under stress if the DSS follows rules of social interaction such as when to interrupt or not (Dorneich et al. 2010). Additionally, students have better learning outcomes when their computer tutor understand the rules of social engagement and politeness (Johnson et al. 2003), and soldiers are more likely to comply in a timely manner to a request made by a DSS when it exhibits an appropriate level of politeness and deference to rank in the phrasing of that request. Likewise, the goal in making polite medication reminder systems, medical robots, and computer health coaches is better patient compliance, and greater long-term adherence to health care and exercise programs.

26.4 CASE STUDIES

Below, case studies are presented focusing on the designs of several quite different DSSs. Within these case studies, a number of the issues raised earlier are discussed in more detail within the context of real-world applications, along with presentations of design solutions that provide concrete illustrations of how to deal with those issues.

26.4.1 Case Study A: Distributed Work in the National Airspace System—Ground Delay Programs

Earlier in this chapter, we indicated that a distributed work system can be characterized in terms of the assignment of control or decision-making authority in relationship to the distribution of knowledge and data. The Federal Aviation Administration's (FAA's) Enhanced Ground Delay Program (GDP) provides a very informative example for illustrating the significance of these parameters when designing a distributed system.

26.4.1.1 Application Area

The FAA's Enhanced GDP has been in operation since 1998. Its goal is to help match the arrival demand at an airport with the available capacity, where demand could exceed capacity because of a weather constraint at that airport, a closed runway, and so on.

The strategy underlying the use of GDPs is to delay departures to the constrained airport, holding these departures on the ground when that airport's arrivals are predicted to exceed capacity. Such a strategy has potential benefits for the FAA in terms of reducing air traffic controller workload (reducing the need for the use of airborne holding or some other air traffic control tactic). It also has potential benefits for the National Airspace System (NAS) Users, as it can be more economical to delay departures, holding these flights on the ground (reducing fuel consumption due to airborne holding as well as reducing the potential for diversions).

26.4.1.2 Original Approach to Distributing Work in a GDP

In principle, there are a variety of different ways to implement GDPs. Prior to 1998, the FAA used GDPs, but did so using an approach that was much less effective than the currently used procedure. Under this original paradigm, traffic managers at the FAA's Air Traffic Control Systems Command Center (ATCSCC) would set the airport arrival rate (AAR) for an airport, and then a DSS would automatically assign an arrival slot to each specific flight, which would then be held on the ground until it was time to depart to make its assigned arrival slot. NAS Users had one opportunity to swap flights in a batch mode (i.e., requesting all of the swaps at one time). This limited their ability to monitor the development of the situation over time and adapt as new information became available. In addition, although NAS Users were asked to cooperate in order to make this program work, there was actually a disincentive for them to provide the FAA with information about cancellations or potential delays. If the cancellation of a flight was reported to the FAA, the NAS User lost its assigned arrival slot; and if it was reported that a flight was going to miss its departure time (because of a mechanical problem, for instance), that flight could be assigned an even greater delay. As a result, NAS Users often chose to withhold this information from the FAA, resulting in unused arrival slots (wasted capacity).

26.4.1.3 Alternative Approach to the Design of GDPs

Given these limitations, a joint industry FAA program evolved to identify and implement a different approach to GDPs. This Collaborative Decision-Making (CDM) program explored methods for still achieving the FAA's primary goal (matching arrival rates to airport capacity), while giving NAS Users the flexibility to better achieve their business goals.

One approach would have been to have NAS Users provide the FAA with all of the knowledge and data about their business concerns relevant to decisions about which flights to assign to which arrival slots. In principle, the FAA traffic managers could have then determined "good" slot allocations for each NAS User, and assigned flights to slots accordingly. Such a decision would have had to somehow take into consideration the competitive nature of this situation and through some decision process determine an "equitable" solution.

It was decided that such an approach would produce unrealistic demands in terms of cognitive complexity and mental workload for these traffic managers, as they would have had to master the relevant expertise of the dispatchers working for each NAS User and to somehow use that knowledge to generate equitable slot assignments. (Dispatchers work for airlines and other general aviation operations that are required to, or choose to, abide by Part 121 regulations of the U.S. Federal Air Regulations, sharing responsibility for each flight with the pilots. They complete the preflight planning for a flight—determining its route, departure time, fuel load, and so on—and are also responsible for following that flight while it is airborne.)

26.4.1.3.1 Strategy for Distributing Work

Instead, the CDM program developed a new architecture for distributing the work associated with GDPs. This architecture uses a classic systems engineering approach for dealing with complexity, decomposing the overall plan for a GDP into a set of nearly independent subplans that can be developed by different individuals and that, when completed by each of these individuals, achieves the desired goals (assuming the plan is successfully implemented). The significance of the qualifier "nearly" should not be overlooked in terms of its importance, however. The reality is that there are cases where interactions among these different individuals are required, so the system design has to include mechanisms that support such interactions in order to ensure that the appropriate perspectives are considered in setting up a GDP, and that exceptions can be made for individual flights when that is required to deal with some overriding consideration.

Like the original GDP, this Enhanced GDP starts with the assumption that a "neutral resource broker" is needed because, in situations where a GDP is called for, the NAS Users are competing for a limited resource (arrival slots at the constrained airport). The critical difference, however, is the application of the following principle:

> The "referee" (the FAA) should only control the process at the level of detail necessary to achieve its goals concerned with safety and throughput, thus allowing NAS Users the

flexibility to try to achieve their own goals to the extent possible subject to this higher level constraint.

Thus, the critical change is the level of abstraction at which ATCSCC (the neutral referee) controls the process. Instead of determining which specific flights are allocated to specific arrival slots, ATCSCC determines the start and stop times for the GDP, the arrival rate for the constrained airport, and the departure airports to be included in the GDP, and then assigns arrival slots to specific NAS Users (such as a specific airline) rather than to specific flights (using a software tool implemented by Metron Aviation called the Flight Schedule Monitor [FSM], shown in Figure 26.1).

The NAS User can then decide for itself which of its flights to assign to one of its arrival slots. An airline might, for instance, have flights from Dallas, Chicago and Miami all scheduled to fly to LaGuardia in New York, all scheduled to arrive between 1800Z–1815Z, and all included in a GDP for LaGuardia. If that airline was given "ownership" of an arrival slot at 1855Z in a GDP for LaGuardia, it could assign that slot to any one of these three flights based on their relative importance to that airline's business concerns—considering factors such as passenger connections, aircraft requirements for later flights out of LaGuardia, crew time restrictions, and aircraft maintenance schedules.

26.4.1.3.2 Ration-by-Schedule

In order to accomplish this new process, there are several requirements. First, a traffic manager at ATCSCC has to set the start and end times and the AAR, and determine which departure airports to include. To do this, ATCSCC solicits input from the relevant FAA facilities and the NAS Users. This interaction takes the form of a teleconference that is held every 2 hours, during which ATCSCC can propose the use of a GDP and provide an opportunity for input from each of the participants. This decision-making process is supported by a simulation or "what-if" function in FSM that allows the user to input a set of parameters (proposed arrival rate, start and end times, and included departure airports) and view the predicted impact in terms of average and maximum delays, and the potential for a spike in demand at the airport after the end time for the GDP. Window 3 in Figure 26.1 shows such a demand pattern for San Francisco (SFO) in 1-hour time bins. The predicted peak demand, for instance, is 45 flights arriving during the 0200Z hour.

After considering these inputs, the ATCSCC traffic manager sets the parameters for the GDP, and lets FSM determine the arrival slot allocations for each NAS User. These allocations are based on "Ration-by-Schedule" (RBS) logic. The number of arrival slots each user gets is proportional to its share of the flights that are included in the GDP that were originally scheduled to arrive at the constrained airport. The slots "owned" by that NAS User are initially assigned to its aircraft based on the order of their originally scheduled arrival times (see Figure 26.2). Thus, if that NAS User originally has 20 flights out of a total of 80 flights that are included in the GDP and are scheduled to arrive at the constrained airport from 1300Z to 1400Z (for a GDP from 1200Z to 1900Z), and if the GDP requires this total of 80 to be reduced to 40, then that NAS User gets 10 slots from 1300Z–1400Z. As a default, the first 10 of its included flights scheduled to arrive after 1300Z are assigned these 10 slots. The arrival times for the remaining 10 flights are pushed back to slots after 1400Z.

Note that, to make this logic work, the GDP needs to cover a time period where, during the GDP or just after its end, there are valleys in the predicted demand that fall below the reduced AAR, so that the program does not create a huge peak in arrival demand at the end of the GDP. The parameters defining the GDP, along with the slot allocations, are then disseminated to the NAS Users.

26.4.1.3.3 Slot Swapping

Abstractly, the strategy of controlling the process at a more abstract level (by setting an AAR) and the strategy of using the RBS algorithm to ration slots to NAS Users are critical,

FSM live data monitor mode

The FSM display consists of four main windows. Clockwise, starting from the upper right, they are as follows:

1. The control window
2. The flight information window
3. The demand graph
4. The timeline

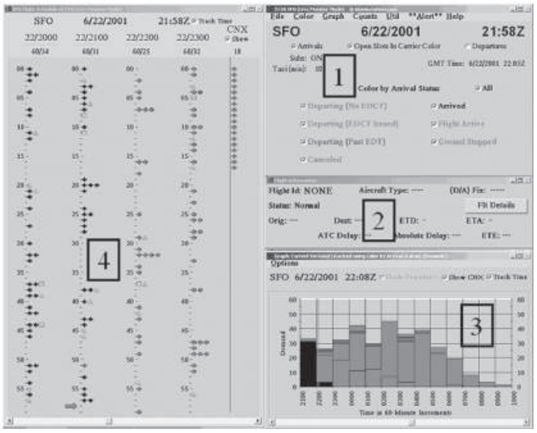


FIGURE 26.1 Screen display from FSM. Window 1 controls the display of information in FSM; Window 2 provides access to information about specific flights included in the GDP; Window 3 provides an indication of the predicted arrival rate, with the display changing based on the parameters set for the GDP, thus supporting "what-iffing"; Window 4 indicates the predicted arrival times for individual flights. (Data from Metron Aviation, Inc.)

as they make it possible to distribute the overall work as relatively independent subtasks. The task of the FAA (the neutral referee) is to set the high-level control parameters. The NAS Users can then work within the resultant constraints to implement plans for their own fleets that are most effective from a business perspective.

The NAS Users accomplish this by swapping flights among assigned arrival slots. Each NAS User can do this by swapping flights among its own slots (see Figure 26.3), or by making a swap with a slot "owned" by another NAS User if such a swap is available.

Traditionally, decisions about what to do with a particular flight were handled by the individual dispatcher responsible for that flight. However, this new process involves making global decisions that consider trade-offs among a number of flights. As a result, instead of requiring some type of potentially time consuming collaboration among the affected dispatchers in that NAS User's Flight Operations Center, the responsibility for managing participation in a GDP has been assigned to a single senior dispatcher (often a new position referred to as the Air Traffic Control [ATC] Coordinator).

Most ATC Coordinators manage the GDP for their fleets using another set of DSSs that allow the ATC Coordinator to

- Test alternative GDPs (GDPs with different parameters) to see which alternative is best from his fleet's perspective. (This information can be used for discussions during the planning teleconference in which the GDP is proposed, providing input to ATCSCC regarding the GDP parameters that should be used.)

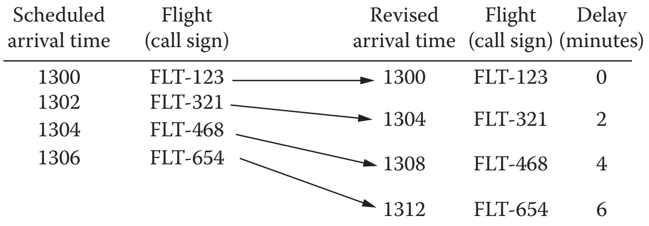


FIGURE 26.2 Arrival slot allocation using the Ration-By-Schedule algorithm. (Assume that the arrival airport can normally handle one arrival every two minutes. If the GDP reduces the arrival rate by 50% for all arrivals, then the flights are assigned revised arrival slots that keep the original ordering, but show increasing delay over time.)

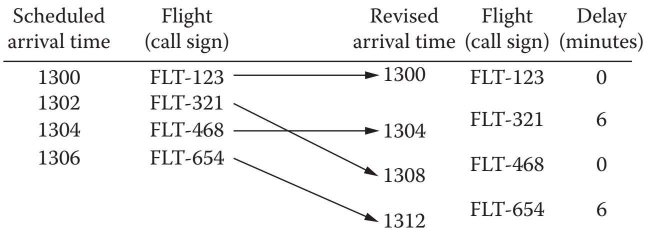


FIGURE 26.3 Arrival slot allocation after swapping FLT-321 and FLT-468. (Note that FLT-468 now has 0 minutes of delay instead of its original 4 minutes of delay.)

- Swap arrival slots among different flights in the fleet either manually or through an automated process. (It is not unusual to make swaps involving a hundred flights based on passenger connections. This requires a DSS that uses an algorithm to estimate the relative importance of the different flights. The ATC Coordinator can, however, constrain or override the solution recommended by the DSS.)
- Test for the impact of strategies for canceling flights or delaying their departure times.

Figure 26.3 illustrates swapping flights belonging to the same NAS User. One very important variation of this feature is canceling a flight and then using its slot to swap with another flight. This is important because this provides the incentive for the NAS User to provide the FAA with critical information:

> According to the "rules of the game" for GDPs (Smith et al. 2003), when the NAS User cancels a flight, that User gets to keep its arrival slot and can swap it just like any other slot under its control. In order to make swaps with canceled flights, however, the User must inform the FAA of the cancellation (accomplished through digital software communications).

This swapping process is further enhanced by a procedure that allows swapping among the different NAS Users. This procedure is referred to as Slot-Credit Substitution.

Slot-credit substitution is best understood in the context of making a decision to cancel a flight. Suppose Airline ABC has flights that arrive every hour on the hour from LGA (LaGuardia Airport) to ORD (Chicago O'Hare). As shown in Figure 26.4, these flights are scheduled to arrive at ORD at 1000Z, 1100Z, 1200Z, and 1300Z.

Suppose a GDP is issued for ORD that delays these flights, assigning revised arrival times (slots) of 1030Z, 1200Z, 1300Z, and 1400Z, respectively. In principle, each of these flights is still capable of departing on time and therefore arriving at its originally scheduled arrival time if a suitable swap could be found to eliminate its delay due to the GDP. Thus, the earliest runway time of arrival (ERTA) for each of these flights is the same as its originally scheduled arrival time. Each flight could, in principle, therefore be swapped to move its arrival to its ERTA. However, it could not be swapped earlier than its ERTA.

Given this situation, the ATC Coordinator for Airline ABC would be willing to cancel Flight 1 (and put the passengers on the next flight) if this would significantly reduce the delay on later flights. In the case shown in Figure 26.4,

Flight	Scheduled arrival time	ERTA(Z)	CTA(Z)	Delay (minutes)
1	1000	1000	1030	30
2	1100	1100	1200	60
3	1200	1200	1300	60
4	1300	1300	1400	60

FIGURE 26.4 Times associated with flights from LGA-ORD with departure delays assigned by a GDP.

however, even if he cancels Flight 1, the slot released is too early for Flight 2 (a flight with an 1100Z ERTA is too late to make use of a slot with a 1030Z CTA). However, there is another airline out there that wants a slot with a 1030Z CTA, and would gladly give up an 1100Z slot to get it. We know this because the ground delay program software detects that another airline has a flight with an 1100Z CTA and has set its ERTA to 1030Z or earlier.

In this case the steps in process work as follows: Airline ABC sends a message saying,

> I am willing to cancel Flight 1 and give up my 1030Z CTA *only if*
>
> a. I can get a slot at 1100Z and put Flight 2 into it
> b. Moving Flight 2 frees up a CTA at 1200Z into which I will put Flight 3
> c. Moving Flight 3 frees up a CTA at 1300Z into which I will put Flight 4.

The ground delay program software looks for a flight to form the necessary bridge and when it finds the one mentioned above, it approves the entire transaction. (If it could *not* find a suitable flight to form the bridge, it would have rejected the entire transaction and Airline ABC would not have been obligated to cancel Flight 1.)

Note that it is not the setting of the ERTA that kicks off the Slot-Credit Substitution process. (ERTAs are always set as a matter of course, by default set as the originally scheduled or filed arrival time for each flight.) Rather, it is the airline's request for a Slot-Credit Substitution transaction that causes the ground delay program software to search for a suitable bridge.

Overall, then, this process has truly distributed the work by changing the locus of control and the parameters of control for certain decisions. The ATCSCC traffic manager's task is to assign the parameters defining the GDP. The ATC Coordinators for each NAS User can then, subject to the constraints imposed by the established GDP, make decisions about which flights to assign to which slots and can even, by setting ERTAs, offer to swap arrival slots with other NAS Users.

26.4.1.3.4 Adapting to Deal with Uncertainty

The GDP process as described thus far basically assumes that the NAS is a deterministic process. If this assumption was true, and if the decision makers had perfect information, then the process would run in a flawless manner, with the right number of flights arriving at the constrained airport at the right time.

However, there are numerous sources of uncertainty in the NAS. The duration and extent of weather constraints are not totally predictable; the estimated time required to clear an aircraft with a blown tire off a runway may be wrong; a flight may miss its assigned departure slot (calculated by working backward from its assigned arrival time and its estimated air time); or a flight may take more or less time than predicted to arrive once airborne. To deal with these uncertainties, certain forms of adaptation have been built into the process.

First, ATCSCC can override the slot allocation for any individual flight, and assign it some other earlier time because of an extenuating circumstance. (In the parlance of the field, such a flight is referred to as a "white hat.") Second, if there are a number of slots before the end of the GDP that have not been filled with active flights, ATCSCC can run a "compression" algorithm that moves later flights up into those slots (subject to the constraints imposed by the ERTAs for those flights and the rule that a flight cannot be moved up to a slot that would require it to depart within the next 30 minutes). Third, ATCSCC can revise the GDP itself because the original predictions or resultant decisions have turned out to be too inaccurate, setting new parameters (such as a new AAR or a new end time) and causing a major shift in arrival slot assignments.

A fourth method for dealing with uncertainty is to pad the original estimate for the AAR. The major reason for doing this is to deal with potential "pop-up" flights. As described earlier, slots are allocated based on the available schedule for arrivals at an airport. However, many general aviation aircraft do not publish a schedule indicating where and when they plan to fly on a given day. They fly on an as-needed basis for their owners. Nevertheless, they still need to be accommodated by the system. To deal with such flights, when setting the AAR using FSM, the ATCSCC traffic manager can indicate the number of pop-ups to plan for, thus leaving some open slots for those flights. When such a flight files its flight plan, it is then assigned the average delay that was assigned to those flights that are now, under the GDP, expected to arrive in the same 15-minute period.

26.4.1.3.5 Administrative Controls

In addition to these real-time decisions, the overall process is monitored for problems. One such potential problem is the quality of the data submitted by a NAS User. To deal with this issue, three metrics are monitored:

- Time-out cancels. A time-out cancel is a flight that is expected to operate (one that was scheduled, but for which a cancellation message was never submitted), but either never operates, or operates well after its assigned departure slot.
- Cancelled-but-flew flights. A cancelled-but-flew flight is a flight that the participant cancels but that ends up operating.
- Undeclared flights. An undeclared flight is a flight belonging to a NAS User that normally submits a schedule of its expected flights for each day, and that operates without prior notice to the system.

NAS Users must demonstrate acceptable performance in terms of these metrics in order to continue to participate in using the slot-swapping functions provided by GDPs. This helps to ensure sufficient data integrity for the overall process to function as planned.

A second area requiring administrative oversight is a concern over gaming. Suppose, for example, that SFO has a GDP in effect because of reduced visibility at that airport. A flight that wants to fly to SFO should take a delay based on this GDP. However, if it files for OAK (Oakland, an airport very close to SFO) instead, and then requests an amendment to land instead at SFO once airborne, it has violated the spirit of the GDP program. ATCSCC monitors for such practices, and takes administrative actions to prevent them in the future.

26.4.1.4 Evaluation of the Implemented Solution

There are two components of the benefits from the Enhanced GDP which has been described above in terms of its major features. The first is the benefit to the NAS Users derived from swapping flights in order to reduce delays for their higher priority flights. These savings are clearly very substantial, but would be very difficult and time consuming to estimate. As a result, to date no study has been conducted to put a dollar figure on that class of benefits.

The second benefit arises from the use of the compression process to fill in unused arrival slots, therefore reducing overall delays. Figure 26.5 shows the frequency of use of the Enhanced GDP process since the introduction of the compression process in 1999. Note that, over the past 2 years, it has not been uncommon to run as many as 100 GDPs per month. Table 26.1 then provides an estimate of the minutes and dollars saved by the compression process alone, using a conservative average cost per minute of operation for a flight of $42/min. These estimates suggest that the use of the compression process alone reduced delay by 28,188,715 minutes from March 18, 1999 to June 30, 2005, thus saving over one billion dollars.

26.4.1.5 Conclusions

The Enhanced GDP illustrates a deliberate effort to design a distributed work system. The design strategy is to have one party (ATCSCC as a neutral referee) set a high-level constraint (the AAR), and to then let the other involved parties (the NAS Users) find solutions that meet their business needs as effectively as possible subject to that constraint (by swapping slots among flights to favor the highest priority flights). Taking this approach, knowledge and data remain distributed, matching the locus of control for the various decisions that need to be made.

TABLE 26.1
Time and Money Savings from the Use of the Compression Process in GDPs for 1999–2005

- Data Included in Analysis:
  - All compression from Mar. 18, 1999–June 30, 2005
  - 3,573,796 flights included
  - 28,188,715 total minutes reduced
  - $1,183,926,030 (@$42/min) dollars saved
- Departure Delay Performance
  - Average reduction: 7.7 minutes
- Percentiles:
  - Reductions greater than 15 minutes: 12.3% (440,889 flights)
  - Reductions greater than 30 minutes: 6.1% (218,886 flights)
  - Reductions greater than 45 minutes: 3.8% (137,424 flights)
  - Reductions greater than 60 minutes: 2.6% (93,558 flights)

Source: Metron Avitation, Inc.

This distribution also serves to reduce the cognitive complexity of the task confronting each individual. The ATCSCC traffic manager does not need to know anything about User priorities, and the ATC Coordinator for a NAS User does not need to become an expert at setting AARs. Finally, because exceptions are necessary to override poor decisions when they are proposed by someone due to a slip or due to insufficient access to the relevant knowledge or data, there must be a mechanism to support interactions among the different participants in order to exchange the necessary information and modify those decisions.

To make this process work, however, certain other requirements arose during its design:

- To make it politically acceptable, nonparticipants (NAS Users who do not choose to participate in slot swapping, etc.) must not be penalized. Such nonparticipants are initially assigned slots just like

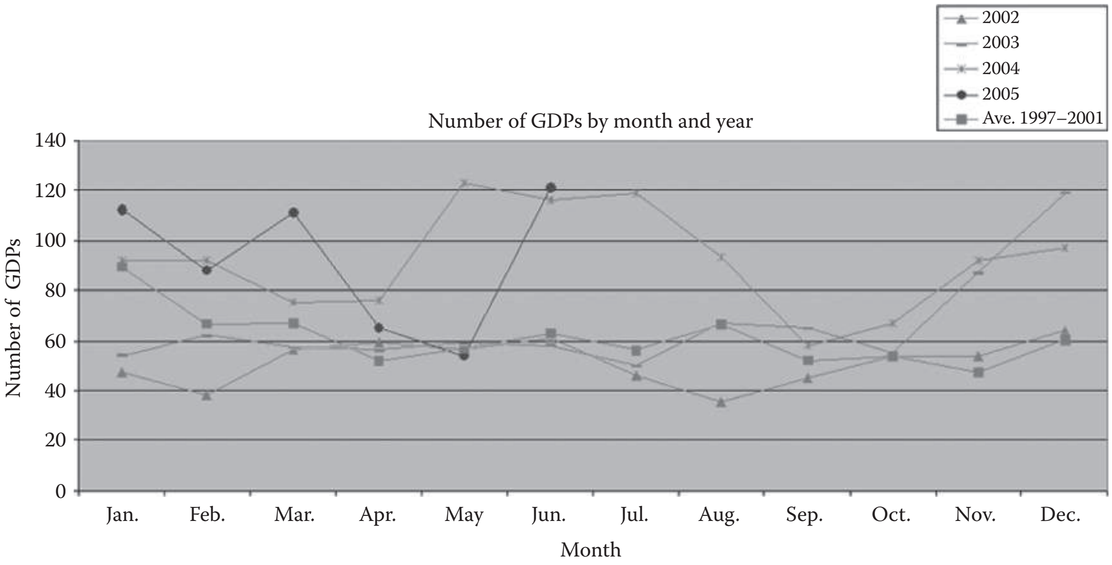


FIGURE 26.5 Frequency of use of the Enhanced GDP by year. (Data from Metron Aviation, Inc.)

everyone else, but their flights are never moved to a later slot as part of any swapping. (They may actually benefit from compression, however, if they are moved up to an earlier slot that is empty.)
- There must be an incentive for information sharing.
- DSSs must be available to provide:
  - Digital communication
  - Simulation of the predicted impact of alternative decisions (what-if capabilities)
  - Algorithmic support for identifying, evaluating and implementing complex decisions involving large numbers of flights (swapping)
  - Information displays that allow the user to understand the situation as relevant to that user's responsibilities, and to effectively evaluate the solution recommended by the technology
  - Control by the user to override the DSS's recommendations in cases where the technology exhibits brittleness
- Organizational changes in roles and responsibilities (new specialists at ATCSCC with the knowledge and skills to develop GDPs; ATC Coordinators for the NAS Users to look at trade-offs in delaying alternative flights).
- The overall procedures must support adaptation, so that decisions can be modified as the scenario evolves.

In summary, this case study serves to illustrate the importance of looking at design from a broad systems perspective when dealing with a distributed work environment, considering not just the detailed design issues necessary to implement a given DSS, but also considering the alternative strategies for distributing the work and the organizational changes required to match people with certain skills and knowledge with corresponding responsibilities, and to make the system resilient in the face of slips, mistakes, and unanticipated events.

26.4.2 Case Study B: Interactive Critiquing as a Form of Decision Support

This case study emphasizes the challenge for interaction designers in dealing with the use of decision-support tools that have the potential for brittle performance due to known or unknown limitations. While there are no perfect solutions to this problem, there are a number of approaches to help reduce its impact.

This case study looks at a tool that incorporates three complementary approaches to the design of an expert system in order to improve overall performance, while reducing the potential impact of brittle performance by the expert system. The first approach to deal with the impact of brittle performance by an expert system is to design a role that encourages the user to be fully engaged in the problem solving, and to apply his knowledge independently without being first influenced by the software. In this case study, the approach explored to achieve this is the use of the computer as a critic, rather than as the initial problem-solver (Fischer et al. 1991; Miller 1986; Silverman 1992; Tianfield and Wang 2004). Thus, instead of asking the person to critique the performance of the software, the computer is assigned the role of watching over the person's shoulder. Note that this is more accurately described as having the design team try to anticipate all of the scenarios that can arise, and then, for all of those scenarios, trying to incorporate the knowledge necessary to detect possible slips or mistakes on the part of the user and to provide alerts and assistance in recovering (Pritchett, Vandor, and Edwards 2002; Wogalter and Mayhorn 2005).

The second approach to reduce sensitivity to brittleness in the computer's performance is to incorporate metaknowledge into the expert system that can help it to recognize situations where it may not be fully competent. By doing so, the software may be able to alert the person to be especially careful because the computer recognizes that this is an unusual or difficult case.

The third approach to deal with brittleness is to develop problem-solving strategies that reduce susceptibility to the impact of slips or mistakes. In the software discussed in this case study, this is accomplished by incorporating a problem-solving strategy that includes the collection of converging evidence using multiple independent problem-solving strategies and sources of data to arrive at a final conclusion.

What follows, then, is a discussion of a specific expert system that incorporates these three strategies for dealing with brittleness. Empirical testing of the system suggests that this approach can significantly enhance performance, even on cases where the software is not fully competent (Smith and Rudmann 2005).

26.4.2.1 Application Area

The specific problem area considered in this case study is the design of a decision-support tool to assist blood bankers in the identification of compatible blood for a transfusion. One of the difficult tasks that blood bankers must complete as part of this process is the determination of whether the patient has any alloantibodies present in his blood serum, and if so, what particular antibodies are present.

26.4.2.1.1 Diagnosis as a Generic Task

Abstractly, this is a classic example of the generic task of abduction or diagnosis (Josephson and Josephson 1994; Psillos 2002). It involves deciding what tests to run (what data to collect), collecting and interpreting those data, and forming hypotheses, as well as deciding what overall combination of problem-solving strategies to employ. Characteristics of this generic task that make it difficult for people include the following:

- The occurrence of multiple solution problems, where more than one "primitive" problem is present at the same time
- The occurrence of cases where two or more "primitive" problems are present, and where one problem masks the data indicative of the presence of the other
- The existence of "noisy" data

- The existence of a large "data space" where data must be collected sequentially, so that the person must decide what data to collect next and when to stop
- The presence of time stress, where an answer must be determined quickly (Elstein, Shulman, and Sprafka 1978; Smith et al. 1998)

In this application, the "primitive" problems are the individual antibodies that may be present in the patient's blood serum. Time stress can arise when the patient is in an emergency situation and needs a transfusion quickly.

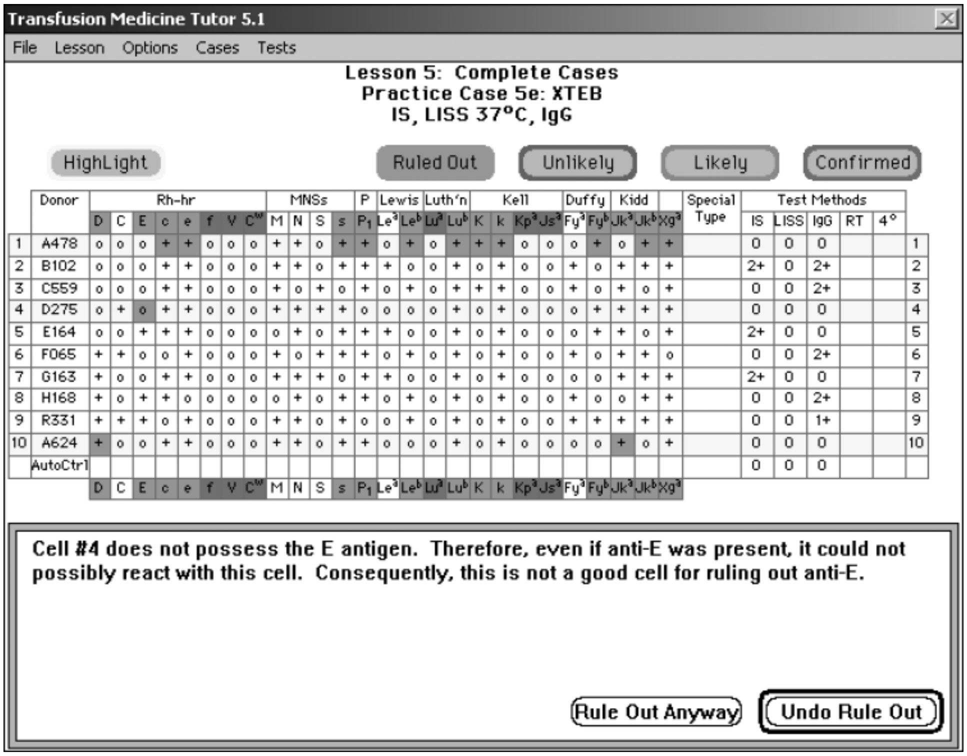
26.4.2.1.2 Sample Problem

To illustrate the nature of this diagnosis task, consider the following partial description of an interaction with the decision-support tool AIDA (the Antibody Identification Assistant) that is the focus of this case study (Guerlain et al. 1999). Initially, the medical technologist needs to determine whether the patient is type A, B, AB, or O, and whether the patient is Rh positive. Then the technologist determines whether the patient shows evidence of any autoantibodies or alloantibodies. As part of this determination, displays like the one shown in Figure 26.6 are provided by AIDA.

To make visual scanning easier on this data display, some of the rows have been highlighted by the technologist in yellow. In addition, to reduce memory load, the technologist has marked a number of intermediate conclusions, indicating that the f, V, Cw, Lua, Kpa, and Jsa antibodies are unlikely (marked in blue in the actual display and in light gray here), and that other antibodies indicated by the labels across the top row can be ruled out (marked in green in the actual display and in dark gray here). These color-coded intermediate answers are transferred to all other data displays to reduce the memory load for the technologist. Figure 26.6 also provides an example of a critique that AIDA has provided in response to an error made by the technologist in marking anti-E as ruled out. This critique was generated by the expert model (rule-based system) underlying AIDA, which monitors all of the test selections and markings made by the technologist as he solves the case using the interface provided by AIDA.

Thus, Figure 26.6 serves to illustrate the following:

- The technologist is responsible for completing the analysis, deciding what tests to run and what intermediate and final conclusions to reach, and is thus very engaged in the task.
- The computer provides an interface that makes it easy for the technologist to select the tests to run and to view the resultant data (using the highlighting to make visual scanning easier), as well as to remember intermediate conclusions (using color-coded markings to indicate these conclusions).
- Although the primary motivation for the technologist in marking the data forms is to make his task easier, when he does so the computer is provided with a great deal of data regarding what the user is thinking and can use those data to make inferences based on its expert model about when to provide a critique cautioning the user that he may have made a slip or mistake.



Lesson 5: Complete Cases
Practice Case 5e: XTEB
IS, LISS 37°C, IgG

	Donor	Rh-hr								MNSs				P	Lewis		Luth'n		Kell				Duffy		Kidd			Special Type	Test Methods					
		D	C	E	c	e	f	V	C^w	M	N	S	s	P$_1$	Le^a	Le^b	Lu^a	Lu^b	K	k	Kp^a	Js^a	Fy^a	Fy^b	Jk^a	Jk^b	Xg^a		IS	LISS	IgG	RT	4°	
1	A478	o	o	o	+	+	o	o	o	+	+	o	+	+	o	+	o	+	+	+	o	o	o	+	o	+	+		0	0	0			1
2	B102	o	o	o	+	+	o	o	o	+	+	o	+	+	+	o	o	+	o	+	o	o	+	o	+	+	+		2+	0	2+			2
3	C559	o	o	o	+	+	o	o	o	+	o	+	o	+	o	+	o	+	+	+	o	o	+	o	+	o	+		0	0	2+			3
4	D275	o	+	o	+	+	o	o	o	+	+	+	+	o	o	o	o	+	o	+	o	o	o	+	+	+	+		0	0	0			4
5	E164	o	o	+	+	+	o	o	o	o	+	o	+	+	+	o	o	+	o	+	o	o	o	+	+	o	+		2+	0	0			5
6	F065	+	+	o	o	+	o	o	o	+	o	+	+	+	o	+	o	+	o	+	o	o	+	o	+	+	o		0	0	2+			6
7	G163	+	o	o	+	+	o	o	o	+	+	+	o	+	+	o	o	+	o	+	o	o	o	o	+	+	+		2+	0	0			7
8	H168	+	o	+	+	o	o	o	o	o	+	o	+	+	o	+	o	+	o	+	o	o	+	o	o	+	+		0	0	2+			8
9	R331	+	+	+	o	+	o	o	o	+	+	o	+	o	o	+	o	+	o	+	o	o	+	+	+	+	+		0	0	1+			9
10	A624	+	o	o	+	+	o	o	o	+	+	o	+	+	o	o	o	+	o	+	o	o	o	o	+	o	+		0	0	0			10
	AutoCtrl																												0	0	0			
		D	C	E	c	e	f	V	C^w	M	N	S	s	P$_1$	Le^a	Le^b	Lu^a	Lu^b	K	k	Kp^a	Js^a	Fy^a	Fy^b	Jk^a	Jk^b	Xg^a							

Cell #4 does not possess the E antigen. Therefore, even if anti-E was present, it could not possibly react with this cell. Consequently, this is not a good cell for ruling out anti-E.

FIGURE 26.6 **(See color insert.)** Full test panel with intermediate results marked using color coded markers provided by AIDA. In this case, AIDA has interrupted the user to suggest that an error may have been made in ruling out anti-E.

26.4.2.1.3 The Need for Decision Aiding

Initial laboratory and field studies indicated that the task of determining the alloantibodies present in a patient's blood is a difficult one for many technologists. A variety of causes of errors were observed, including slips, perceptual distortions, incorrect or incomplete knowledge (Guerlain et al. 1999), and cognitive biases (Brachman and Levesque 2004; Bradfield and Wells 2005; Fraser, Smith, and Smith 1992; Gilovich, Griffin, and Kahneman 2002; Haselton, Nettle, and Andrews 2005; Kahneman and Tversky 2000; Plous 1993; Poulton 1989).

26.4.2.2 Design Solution

It was the judgment of the design team that this task was sufficiently complex that, given the available development resources, it was unlikely that all of the possible scenarios could be anticipated and dealt with by the design team. It was also noted that, for this task, the cost of an erroneous diagnosis was potentially high. Thus it was decided that, instead of automating the task, a DSS should be developed that kept the person very much engaged in the task, and that provided other safety nets to reduce the chances of error for those cases where the computer's knowledge was incomplete. This conclusion was reinforced by a preliminary study regarding the impact of role on diagnostic accuracy in antibody identification. Guerlain et al. (1996) showed that, when the user was asked to critique the computer's performance instead of having the computer critique the user's performance, the final answer was wrong 31% more often on cases where the computer's knowledge was incomplete when, on that case, the person was assigned to the role of critic rather than having the computer critique the person.

26.4.2.2.1 Critiquing—Additional Design Considerations

The literature provided additional guidance in deciding whether and how to develop this software to play the role of a critic. For example, Miller (1986) developed a prototype system called ATTENDING that focused on anesthesiology. Based on studies of its use, Miller suggested that critiquing systems are most effective in applications where the user has a task that is frequently performed, but that requires the person to remember and apply a great deal of information in order to complete a case. Miller's conclusion was that, on such tasks, the person is more susceptible to slips and mistakes and would therefore benefit significantly from the DSS.

A second consideration was how intrusive the interactions with the knowledge-based system would be for the user. A number of researchers have suggested that an interface that requires that the user spend significant time entering data and informing the computer about what he has concluded is too cumbersome, and will therefore be unlikely to be adopted in actual practice (Berner et al. 1989; Harris and Owen 1986; Miller 1984; Shortliffe 1990; Tianfield and Wang 2004).

A third consideration was concern over the potential for complacency if the person played the role of critic, letting the computer complete an initial assessment and then having the person decide whether to accept this assessment. Parasuraman and Riley (1997) have shown that, in such a role, there is a risk that the person will become over reliant on the computer, and will not adequately apply his knowledge in completing the critique (Metzger and Parasuraman 2005). (Note, however, that a person could become over reliant even with the roles reversed, as the person might start to get careless and assume the computer will always catch his slips. Administrative controls, based on regular monitoring of the person's performance, might help reduce such complacency, but this is as yet an unexplored aspect regarding the use of critiquing systems.)

A final consideration was the timeliness of critiques. Waiting for the user to reach a final answer before providing a critique, as has been the case for many critiquing systems, is potentially inefficient and objectionable, as the user may have invested considerable time and effort in arriving at a mistaken answer that the computer could have headed off earlier in the person's problem-solving process. Furthermore, if the critique is given well after the user has made a slip or mistake, it may be difficult for him to remember exactly why he arrived at that intermediate conclusion (thus making it harder to decide whether to accept the computer's critique). This consideration therefore suggests the need for an interface that provides the computer with data about the user's intermediate conclusions rather than just the user's final answer, so that critiquing can be more interactive.

Based on these considerations, AIDA was developed as a critiquing system that supported the user as he completed the antibody identification task. In order to provide immediate, context-sensitive critiques, an interface was developed that encouraged the user to mark intermediate conclusions on the screen. As suggested earlier, these markings in fact reduced the perceptual and memory loads for the user, thus implicitly encouraging this form of communication and allowing the computer to detect and respond immediately with context sensitive critiques to potential slips and errors made by the person.

26.4.2.2.2 Complementary Strategies to Reduce Susceptibility to Brittleness

The considerations outlined above focused on how to keep the person engaged in the diagnosis task, and how to avoid triggering the cognitive biases that can arise if the computer suggests an answer before the person has explored the data himself (Larson and Hayes 2005; Smith, McCoy, and Layton 1997). AIDA also incorporated two additional design strategies to reduce the potential for error. One was the incorporation of metaknowledge into the knowledge-based system. The other was to include a problem-solving strategy that was robust even in the face of slips or mistakes by either the person or the computer (i.e., the design team).

Metaknowledge was included to help the computer identify cases where its knowledge might be incomplete. Such rules were developed by identifying the potential weak points in the computer's problem-solving process. An example was the use of thresholding by AIDA when applying rules such as the following:

If a test cell's reactions are 0 on a test panel for all three of the testing conditions (IS, LISS and IgG) as they are for the first test cell (Donor A478), shown in Figure 26.6, and if e is present (shown by a + in the column labeled e in the row corresponding to the first test cell) on that test cell but E is not (shown by a 0 in the column labeled E in the row corresponding to the first test cell), then mark e as ruled out.

This heuristic usually produces the correct inference. Since it does not directly reason with some form of probabilistic reasoning (Bochman 2004; Pearl 1988; Shafer and Pearl 1990; Shafer 1996), however, it can lead to ruling out an antibody that is actually present in the patient's serum with no consideration of the level of uncertainty. This is most likely to happen when the reaction strengths are weak. Thus, AIDA was provided with a rule that monitored for weak reactions on cells, and when this situation was detected and the user was observed trying to complete rule-outs without first enhancing the reactions with some alternative test phase, the system cautioned the user. In this way, AIDA put the user on alert to be especially careful in applying his normal strategies and rules.

A second strategy incorporated into AIDA as protection against brittleness was to monitor for the collection and consideration of converging evidence by the technologist. This problem-solving strategy was observed in use by one of the experts involved in the development of AIDA. She used this strategy to catch her own errors and those of the technologists working under her supervision. The strategy was based on the assumption that any one data source or line of reasoning could be susceptible to error, and that it is therefore wise to only accept a conclusion only if independent types of data (test results that are not based on the same underlying data or process) and independent sets of heuristics or problem-solving strategies have been used to test that conclusion. Based on this expert strategy, AIDA monitored the user's problem-solving process to see whether such converging evidence had been collected prior to reaching a final answer. If not, AIDA cautioned the user and suggested types of converging evidence that could be collected.

26.4.2.3 Evaluation of the Implemented Solution

From the standpoint of HCI, the key question is how effectively this human–machine system performs using this cooperative problem-solving paradigm (Erkens, Andriessen, and Peters 2003). To gain insights into this, an empirical study was conducted using AIDA.

This study of 37 practitioners at 7 different hospitals found that those blood bankers using AIDA with its critiquing functions turned on made significantly fewer errors ($p < .01$) than those that used AIDA as a passive interface. Errors in the final diagnoses (answers) were reduced by 31%–63%.

On those cases where AIDA was fully competent (posttest Cases 1, 2, and 4), errors (percentage of technologists getting the wrong final answer) were reduced to 0% when the critiquing functions were on (treatment group). On the case where AIDA was not fully competent (posttest Case 2), AIDA still helped reduce errors by 31% (Table 26.2).

These empirical results support the potential value of interactive critiquing, supplemented with the metaknowledge and problem-solving strategies that were embedded in the software, as a design strategy for applications where concerns regarding the potential brittleness of the technology are deemed significant.

TABLE 26.2
Percentage of Blood Bankers Arriving at the Wrong Final Answer on Four Test Cases

	Control Group (%)	Treatment Group (%)
Posttest case 1	33	0
Posttest case 2	50	19
Posttest case 3	38	0
Posttest case 4	63	0

26.4.3 Case Study C: Impact of a Decision-Support System on Individual versus Group Decision Making

This case study emphasizes how a DSS may have very different effects on an individual problem solver verses a group working jointly. System designers should not assume that groups will gain the same benefits from a DSS as individuals, or vice-versa.

The domain examined in this case study is military planning, specifically enemy course of action (ECOA) planning. In this task, small groups of planners (or intelligence analysts) examine intelligence information gathered from scouts in the field, radio traffic, sensors, satellite data and other sources and try to form hypotheses about what possible and likely plans of action the enemy forces may take. Each of these plans is called an ECOA. Friendly forces then use the ECOAs, which are their best guesses as to what the enemy may do, to develop a set of friendly courses of action (FCOA) that will allow them to respond to all the likely and dangerous actions which the enemy may pursue.

26.4.3.1 Decision-Support System to Assist Military Planners

This case study examines Weasel, a DSS designed to assist military planners in systematically identifying all ECOAs based on the most likely positions of resources and activities identified by intelligence (Ravinder 2003; Hayes and Ravinder 2003). Note, that Weasel does not find all possible ECOAs, just the most likely based on intelligence.

Weasel interacts with the user in the role of an intelligent planning assistant. Users do their own planning and make their own judgments, but they can ask Weasel to suggest ECOAs. Users can then decide which ECOAs to include in their own solution sets. They can also modify Weasel's ECOAs.

Figure 26.7 shows an example of an ECOA. Each rectangle represents a military unit. The large arrows represent the direction of movement of the attacking forces, and roughly

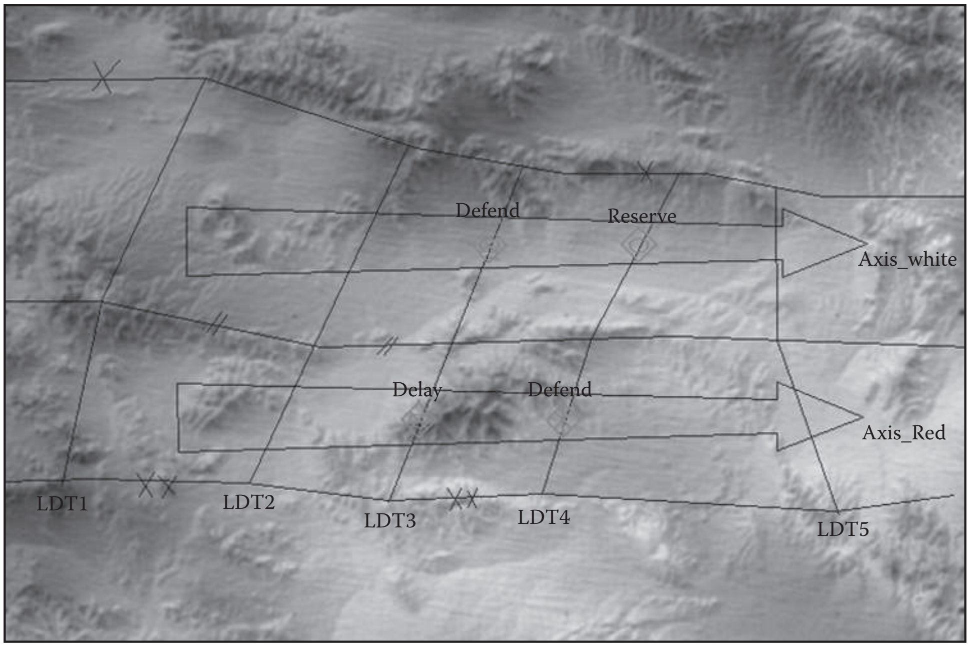


FIGURE 26.7 (See color insert.) An example of an enemy course of action (ECOA). The diamond-shaped figures represent the enemy units. Each unit is further labeled by role played by the unit in this scenario: Defend, Reserve, or Delay. The enemy units move along two corridors in the terrain indicated by large arrows and labeled Axis White and Axis Red. Friendly units move in the direction of the arrows, enemy units move in the reverse direction. LDT means Line of Defensible Terrain. LDT 1 through 5 are used as reference lines to mark how far the units have advanced. In this snapshot, all enemy units are currently positioned on LDT 3 and LDT 4.

mark out the area in which units will move. The ECOAs generated by Weasel are at a very high level of abstraction, specifying primarily the current positions of each enemy unit or resource, but not all the steps in the plan. This is a standard abstraction used by the military to represent ECOAs before much of the detail is known; each ECOA is essentially a "shorthand" for a whole family of more detailed ECOAs that might arise from that plan.

While ECOA planning is couched as a planning task, it is in many ways very similar to a diagnostic task (Simmons 1988). The intelligence information gathered is analogous to the observations made by medical personnel and the result of diagnostic tests ordered. And, as in diagnosis:

- The data may be noisy.
- The data space is very large.
- There is time pressure to develop a set of ECOAs quickly, and dangerous consequences for bad decisions. In military situations, lives may be lost if decisions are not timely and on target.
- There may be many possible explanations for the observed data. In diagnosis one problem may mask another. In ECOA planning, the enemy may take one set of actions to trick the opposition into believing their plan is different than it really is.

As it is impossible to know for certain what the enemy is really planning, the job of an ECOA planner is to identify several ECOAs that he or she deems as likely, or especially dangerous even if unlikely.

By planning for multiple possibilities, the friendly forces can prepare in advance for whatever the enemy may do, and avoid being taken by surprise. However, in practice, it is often difficult for people to systematically think of all the possibilities, of all the many ways in which the enemy may distribute its forces and allocate its resources. By offering a DSS to perform a systematic and thorough search of the solution space, the goal is to combine computers' thoroughness, with human judgment and flexibility to produce a system that produces better results than either alone.

26.4.3.2 As an Assistant for Individuals

Larson and Hayes (2005) conducted a study in which they assessed whether using Weasel had an impact on the quality of ECOAs produced by individual military planners working alone. Quality was judged independently by two expert military planners. Eighteen subjects participated in the experiment (9 Air Force and 9 Army subjects). The average length of experience across all 18 subjects was 5.03 years. Five subjects were categorized as domain experts, and 13 as intermediates. Experts were those having 6 or more years of military experience on active duty, in the National Guard or Reserves.

Each subject was individually asked to generate a set of ECOAs which they would pass on to their commander, for various scenarios. They did these tasks without Weasel's assistance and with Weasel's assistance. Scenario, method, and order were systematically varied. The two expert judges each independently assigned quality scores to the final set of ECOAs produced by each individual. The results demonstrated that Weasel significantly improved performance for individuals ($p = .018$).

26.4.3.3 As an Assistant for Groups

Larson repeated this experiment five years later, but this time he additionally examined the impact of Weasel on *groups* of planners working together to generate ECOAs, which is more representative of how ECOA planning is done in practice (Larson 2010). Larson studied 12 groups, each of which had 3 members, one of whom played the role of group leader. The group leader was assigned randomly. As in the experiment above, the groups were asked to create sets of ECOAs which they would pass on to their commander without the aid of Weasel, and with the aid of Weasel. Their ECOA sets were independently given quality scores by two different expert evaluators, who were different individuals than those that participated in the previous experiment. The evaluators also independently assigned quality scores to ECOA sets produced by 18 individuals. Again, Weasel significantly improved the quality of the ECOA sets produced by individuals, but not by groups. The best predictors of solution quality for groups were not whether or not they used Weasel, but the total number of ECOAs produced in order to develop the solution set (even though most of those ECOAs did not end up in the final solution set), and the amount of interaction between the group members.

A closer look at the data provides insights into these results. First of all, teams and individuals earned almost the same average solution quality scores when they used Weasel. However, without Weasel, teams typically performed better than did individuals. Thus, Weasel brought the performance of individuals up further than the performance of teams.

An examination of the interactions between group members provides additional insights into why the teams performed better than individuals without Weasel; group mates provided error checking on others' solutions, and produced alternative ECOAs which others did not think of. High-performing groups produced more ECOAs overall en route to a solution because multiple "heads" were able to think up a larger numbers of distinctly different ECOAs than could any one individual, and more ECOAs were associated with higher quality solutions. Furthermore, the high performing groups often worked together to make notations on the same sheet of paper, or looked over the shoulder of another teammate while commenting on the solutions. In addition, the "technical lead" of the group was not always the assigned group leader.

In contrast, groups that performed poorly communicated less frequently with each other. In the videotapes of the groups, one can see the lack of connection in the body language of the poorly performing groups. Each sat side by side working on his or her own sheet, or doing nothing. Occasionally someone would speak but the others might not answer or even move in response.

These insights may jointly provide an explanation as to why Weasel provided more benefit to individuals than groups. It appeared that for individuals, the Weasel DSS played the role of an additional group member by providing an alternative perspective and additional ECOAs that the individual would not have thought of on his or her own. However, the groups already had two other group mates to perform the role which the DSS played for the individuals. Thus, it did not provide as much of a performance "boost" to the teams; they already got that boost from their team mates.

26.4.3.4 Lessons Learned

The main take away point of these studies for the DSS designer is that groups may not derive the same benefits that individuals derive from a DSS, particularly if the DSS is designed to play the role of an intelligent assistant or additional team mate. This is particularly important in work that is typically carried out by groups. If one designs a DSS and demonstrates that it improves performance for individuals, one cannot assume that it will provide the same benefits (or any benefits) when used in practical situations by groups.

26.4.4 Case Study D: Rules of Social Interaction—Interruptions and Their Uses

This case study provides some insights into how designing DSSs to follow culturally expected norms for interaction can improve performance benefits of DSSs. This case study examines rules for when to interrupt. This is important because it is usually inappropriate to interrupt someone in the middle of an important task and it can be dangerous to interrupt a life-critical task (Dorneich et al. 2010). People are likely to become irritated with "rude" DSSs that interrupt at inappropriate and inconvenient times, and turn them off. However, it can be equally inappropriate and dangerous to fail to interrupt in a timely manner with critical information.

Dorneich et al. (2010) performed two studies of DSSs to assist dismounted soldiers in life-critical tasks. The first system, the Communications Scheduler, reduced soldiers' cognitive workload during high workload periods by converting low priority radio messages into text which was displayed on a handheld display carried by the soldier. Examples of high workload periods included times when the soldier was executing a complex maneuver in an urban battle environment, or when the soldier was avoiding gunfire. Essentially, the Communications Scheduler reduced interruptions during high workload times so that the soldier could concentrate on the task.

The second system, the Tactile Navigation Cueing System, delivered navigation directions to the dismounted soldiers by "prodding" them with vibrating pads attached to a belt (called a tactor belt) worn around the waist. The direction of the vibration on the belt indicated the direction in which the soldier needed to travel in order to avoid (simulated) landmines in a field. Thus, the Tactile Navigation Cueing System did the opposite of the Communications Scheduler; it added interruptions, rather than reducing them, but its interruptions provided safety-critical information intended to keep the soldier from stepping on a mine.

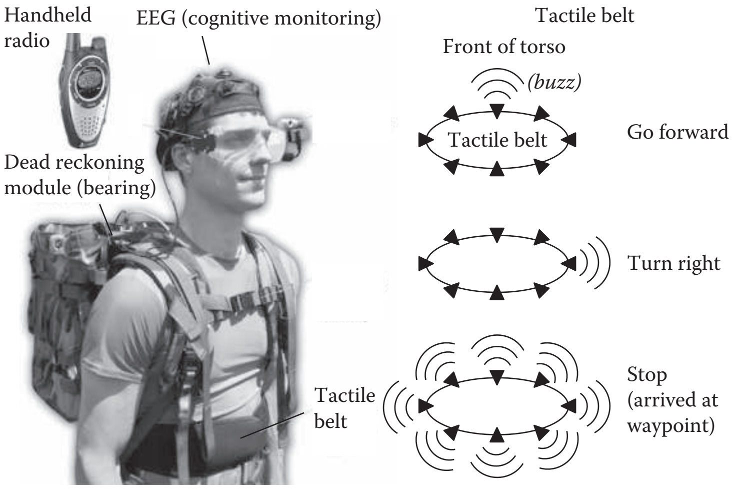


FIGURE 26.8 Equipment for the communications scheduler and tactile navigation cueing systems.

26.4.4.1 Experimental Setup

The setup for both experiments was very similar, and is shown in Figure 26.8, the difference being that the Communications Scheduler did not use a tactor belt. Both systems are portable and mounted on the body so that dismounted soldiers can carry the system along with them in the field. The computer was carried in a backpack, and sensors and other equipment were mounted on the head and body. Both systems monitor the soldier's cognitive state using EEG (brain activity) and ECG (heart) sensors. The EEG sensors are incorporated into a special cap that participants wore. The ECG sensors were glued to the torso with a medical adhesive patch. All sensors were connected wirelessly to the computer in the backpack. These sensors are used to assess the soldier's cognitive workload (high or low). Soldiers also carried a radio clipped to their shoulders, and a handheld display.

26.4.4.2 Reducing Low-Priority Interruptions

Eight subjects tested the Communication scheduler. Each was asked to perform multiple tasks while navigating outdoors in a partially wooded field used as a mock battle environment, including monitoring radio traffic, keeping a count of the number of civilians and soldiers sighted.

The complexity of the navigation route and the frequency of radio messages were varied to produce high workload and low workload situations. When the Communications Scheduler detected that the soldier was experiencing a high workload (through the EEG and ECG sensors), it converted radio messages pertaining to low-priority tasks into text messages, allowing only the high-priority radio messages through, thus reducing the total number of interruptions.

Subjects were each tested in scenarios in a 2 × 2 within subjects design in which they performed multiple simultaneous tasks without the Communications Scheduler being available to them, and with the Communications Scheduler, under both high and low workload conditions.

The results showed that use of the Communications Scheduler significantly improved soldiers' accuracy during high workload periods for the high-priority tasks, for example, maintaining counts of civilians and soldiers sighted ($p < .05$) and monitoring the maneuvers of their own unit ($p < .05$). There was no significant impact of the Communications Scheduler during low workload periods.

However there was also a cost associated with using the Communications Scheduler. While it reduced interruptions, therefore allowing soldiers to concentrate on the high-priority tasks, they rarely went back to read all the lower priority (but still important) text messages that had been taken out of the radio traffic stream and sent to their handheld displays. This resulted in reduced situation awareness as they did not absorb that information.

26.4.4.3 Increasing Safety-Critical Interruptions

Six subjects tested the Tactile Navigation Cueing System. Subjects used the same equipment as the experiment above, plus a tactor belt to navigate the same outdoor environment (a partially wooded field). In this study, subjects were also asked to perform multiple tasks simultaneously: maintain counts of civilians and soldiers sighted, monitor the maneuvers of their own unit, and perform math tasks. However, their primary task was to navigate a complex path through the field, while avoiding sites representing landmines and surveillance cameras.

Subjects were asked to perform these simultaneous tasks in a 2 × 2 within subjects design, without and with the availability of the Tactile Navigation Cueing System, under high and low workload conditions. High workload conditions were created by making the path through the mine field more complex, and the radio messages more frequent.

The results revealed that subjects were able to complete the navigation task more quickly, running into fewer (pretend) mines and surveillance cameras, when using the Tactile Navigation Cueing System. Despite the additional interruptions from the tactor belt, they also increased their accuracy in the math task, while accuracy in other tasks remained unchanged.

This may show that, by allowing the DSS to take over the navigation task so they did not have to think about which way to go anymore, it freed up cognitive resources for other tasks. However, as in the previous experiment, the increased performance came at the cost of situation awareness. One soldier's attention was so focused on the vibrating tactor belt and its directions that he forgot to attend to what was around him and he ran into a low hanging tree branch.

26.4.4.4 Lessons Learned

DSSs that do not follow appropriate rules for interaction, for example by interrupting at inappropriate times, often irritate users and may result in disuse of that system. This case study demonstrates that designing a DSS to follow appropriate social rules for interruption can improve performance. Reduced interruptions during high workload periods can improve performance, as can appropriate interruptions to deliver critical information.

However, even when interruptions are appropriately filtered or delivered, they come at a cost of which DSS designer should be aware. Use of such strategies may reduce situation awareness, a cost which must be carefully balanced with the DSS's benefits.

26.5 CONCLUSION

One goal of this chapter has been to outline different conceptual approaches to support the design of DSSs, including the use of CTAs and work domain analyses. A second goal has been to review relevant knowledge about the psychologies of both the designers and users of DSSs, identifying important design questions that need to be considered in the development of such software systems. Finally, the chapter emphasizes the need to take a broader systems perspective when deciding how to integrate DSSs into a work environment, as there are a number of other high-level design parameters (such as deciding how to distribute the work among a team of people and machines) that may offer alternative opportunities for effectively incorporating DSSs into the overall system.

REFERENCES

Amalberti, R. 1999. Automation in aviation: A human factors perspective. In *Handbook of Aviation*, ed. D. Garland, J. Wise, and V. Hopkins, 173–92. Mahwah, NJ: Lawrence Erlbaum.

Anderson, J. R. 1993. *Rules of the Mind*. Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R., D. Bothell, N. Byrne, S. Douglass, C. Lebiere, and Y. Qin. 2004. An integrated theory of the mind. *Psychol Rev* 111(4):1036–60.

Andes, R. C., and W. B. Rouse. 1992. Specification of adaptive aiding systems. *Inf Decis Technol* 18:195–207.

Annett, J., K. D. Duncan, R. B. Stammers, and M. J. Gray. 1971. *Task Analysis*. London, UK: HMSO.

Annett, J., and N. Stanton. 2000. *Task Analysis*. London: Taylor & Francis.

Baddeley, A. 1998. *Human Memory: Theory and Practice*. Boston, MA: Allyn and Bacon.

Baddeley, A. 1999. *Essentials of Human Memory*. London: Taylor & Francis Group.

Baecker, R., J. Grudin, W. Buxton, and S. Greenberg. 1995. *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration*. San Francisco, CA: Morgan Kaufmann.

Bainbridge, L. 1983. Ironies of automation. *Automatica* 19:775–9.

Bar-Yam, Y. 2003. *Dynamics of Complex Systems: Studies in Nonlinearity*. Nashville, TN: Westview Press.

Bar-Yam, Y. 2005. *Making Things Work: Solving Complex Problems in a Complex World*. Cambridge, MA: NECSI Knowledge Press.

Beach, L. R., and T. Connolly. 2005. *The Psychology of Decision Making: People in Organizations*. 2nd ed. Thousand Oaks, CA: Sage Publications, Inc.

Bennett, K. B., and J. M. Flach. 1992. Graphical displays: Implications for divided attention, focused attention, and problem solving. *Hum Factors* 34(5):513–33.

Berner, E., C. Brooks, R. Miller, F. Masarie, and J. Jackson. 1989. Evaluation issues in the development of expert systems in medicine. *Eval Health Prof* 12:270–81.

Bickmore, T. 2010. Etiquette in motivational agents: Engaging users and developing relationships. In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 205–29. Boca Raton, FL: Taylor & Francis.

Billings, C. E. 1997. *Aviation automation: The search for a human-centered approach*. Hillsdale, NJ: Lawrence Erlbaum.

Bisantz, A. M., E. Roth, and B. Brickman. 2003. Integrating cognitive analysis in a large-scale system design process. *Int J Hum Comput Stud* 58:177–206.

Bittner, K. 2002. *Use Case Modeling*. Harlow, England: Addison-Wesley.

Bjork, R. A. 1999. Assessing our own competence: Heuristics and illusions. *Atten Perform* 17:435–59.

Blank, H. 1998. Memory states and memory tasks: An integrative framework for eyewitness memory and suggestibility. *Memory* 6(5):481–529.

Blount, S., and R. P. Larrick. 2000. Framing the game: Examining frame choice in bargaining. *Organ Behav Hum Decis Process* 81(1):43–71.

Bochman, A. 2004. A causal approach to nonmonotonic reasoning. *Artif Intell* 160(1–2):105–43.

Bodker, K., F. Kensing, and J. Simonsen. 2004. *Participatory IT Design: Designing for Business and Workplace Realities*. Cambridge, MA: MIT Press.

Bowers, C., E. Salas, and F. Jentsch, eds. 2006. *Creating High-Tech Teams: Practical Guidance on Work Performance and Technology*. Washington, DC: American Psychological Association.

Brachman, R., and H. Levesque. 2004. *Knowledge Representation and Reasoning*. San Mateo, CA: Morgan Kaufmann.

Bradfield, A., and G. Wells. 2005. Not the same old hindsight bias: Outcome information distorts a broad range of retrospective judgments. *Mem Cognit* 33(1):120–30.

Brehm, S., S. M. Kassin, and S. Fein. 1999. *Social Psychology*. 4th ed. Boston, MA: Houghton Mifflin Company.

Burns, C., A. Bizantz, and E. Roth. 2004. Lessons from a comparison of work domain models: Representational choices and their implications. *Hum Factors* 46:711–27.

Burns, C., and J. Hajdukiewicz. 2004. *Ecological Interface Design*. Boca Raton, FL: CRC Press.

Burns, C., and K. Vicente. 2001. Model-based approaches for analyzing cognitive work: A comparison of abstraction hierarchy, multi-level flow modeling, and decision ladder modeling. *Int J Cognit Ergon* 5:357–66.

Byrne, M., and A. Kirlik. 2005. Using computational cognitive modeling to diagnose possible sources of aviation error. *Int J Aviat Psychol* 15(2):135–55.

Caggiano, D., and R. Parasuraman. 2004. The role of memory representation in the vigilence decrement. *Psychon Bull Rev* 11(5):932–7.

Caldwell, B. 2005. Multi-team dynamics and distributed expertise in mission operations. *Aviat Space Environ Med* 76(6):145–53.

Card, S., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum.

Carroll, J. M. 1995. *Scenario-Based Design: Envisioning Work and Technology in System Development*. New York: John Wiley & Sons.

Carroll, J. M., D. C. Neale, P. L. Isenhour, and D. S. McCrickard. 2003. Notification and awareness: Synchronizing task-oriented collaborative activity. *Int J Hum Comput Stud* 58:605–32.

Cartwright, D., and A. Zander, eds. 1960. *Group Dynamics: Research and Theory*. 2nd ed. Evanston, IL: Row and Peterson.

Cary, M., and R. Carlson. 2001. Distributing working memory resources during problem solving. *J Exp Psychol: Learn Mem Cogn* 27:836–48.

Chandrasekaran, B. 1988. Generic tasks as building blocks for knowledge based systems: The diagnosis and routine design examples. *Knowl Eng Rev* 3(3):183–210.

Charness, N., P. Feltovich, R. Hoffman, and E. Ericsson. 2006. *The Cambridge Handbook of Expertise and Expert Performance*. New York: Cambridge University Press.

Chi, M., R. Glaser, and M. Farr, eds. 1988. *The Nature of Expertise*. Hillsdale, NJ: Lawrence Erlbaum.

Clancey, W. J. 1983. The epistemology of a rule-based expert sytem—a framework for explanation. *Artif Intell* 20:215–51.

Clancey, W. J. 1985. Heuristic classification. *Artif Intell* 27:289–350.

Clarke, K., G. Hardstone, M. Rouncefield, and I. Sommerville, eds. 2006. *Trust in Technology: A Socio-Technical Perspective*. New York: Springer.

Cockburn, A. 2000. *Writing Effective Use Cases*. Harlow, England: Addison-Wesley.

Crandall, B., G. Klein, and R. Hoffman. 2006. *Working Minds: A Practitioner's Guide to Cognitive Task Analysis*. Cambridge, MA: MIT Press.

Croson, R., and J. Sundali. 2005. The gambler's fallacy and the hot hand: Empirical data from casinos. *J Risk Uncertain* 30(3):195–209.

Dasgupta, P., P. Chakrabarti, and S. Desarkar. 2004. *Multiobjective Heuristic Search: An Introduction to Intelligent Search Methods for Multicriteria Optimization*. San Mateo, CA: Morgan Kauffmann.

Dechter, R. 2003. *Constraint Processing*. San Mateo, CA: Morgan Kaufmann.

Diaper, D., and N. Stanton, eds. 2004. *The Handbook of Task Analysis for Human-Computer Interaction*. Mahwah, NJ: Lawrence Erlbaum.

Dorneich, M. C., S. Mathan, S. Whitlow, P. M. Ververs, and C. C. Hayes. 2010. Etiquette considerations for adaptive systems that interrupt: Costs and benefits. In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 289–319. Boca Raton, FL: Taylor & Francis.

Eckstein, M. P. 2004. Active vision: The psychology of looking and seeing. *Perception* 33(8):1021–3.

Elstein, A. S., L. S. Shulman, and S. A. Sprafka. 1978. *Medical Problem Solving: An Analysis of Clinical Reasoning*. Cambridge, MA: Harvard University Press.

Endsley, M. 2003. *Designing for Situation Awareness*. London: Taylor & Francis.

Endsley, M., and D. Garland. 2000. *Situation Awareness Analysis and Measurement*. Mahwah, NJ: Lawrence Erlbaum Associates.

Erkens, G., J. Andriessen, and N. Peters. 2003. Interaction and performance in computer-supported collaborative tasks. In *Cognition in a Digital World*, ed. H. Van Oostendorp, 225–51. Mahwah, NJ: Lawrence Erlbaum.

Flach, J., P. Hancock, J. Caird, and K. Vicente, eds. 1995. *Global Perspectives on the Ecology of Human-Machine Systems*. Hillsdale, NJ: Lawrence Erlbaum.

Fischer, G., A. C. Lemke, T. Mastaglio, and A. I. Morch. 1991. The role of critiquing in cooperative problem solving. *ACM Trans Inf Syst* 9(3):123–51.

Forsyth, D. 1998. *Group Dynamics*. Florence, KY: Wadsworth Publishing.

Fraser, J. M., P. J. Smith, and J. W. Smith. 1992. A catalog of errors. *Int J Man Mach Syst* 37:265–307.

Freeman, F., P. Mikulka, and M. Scerbo. 2004. An evaluation of an adaptive automation system using a cognitive vigilance task. *Biol Psychol* 67(3):283–97.

Garb, H. 2005. Clinical judgment and decision making. *Annu Rev Clin Psychol* 1(1):67–89.

Geddes, N. D. 1989. *Understanding Human Operator's Intentions in Complex Systems*. Doctoral Dissertation, Georgia Institute of Technology, Atlanta, GA.

Geddes, N. D. 1997a. Large scale models of cooperating and hostile intents. In *Proceeding of the 1997 IEEE Conference on Engineering of Computer Based Systems*. New York: IEEE.

Geddes, N. D. 1997b. Associate Systems: A framework for human-machine cooperation. In *Design of computing systems: Social and ergonomic considerations. Advances in Human Factors/ Ergonomics*, ed. M. J. Smith, G. Salvendy, and R. J. Koubek, Vol. 21, 237–42. New York: Elsevier.

Geddes, N. D., and C. S. Lizza. 1999. Shared plans and situations as a basis for collaborative decision making in air operations. *SAE World Aeronautics Conference*, SAE Paper 1999-01-5538.

Gentner, D., and A. Stevens. 1983. *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum.

Ghallib, M., D. Nau, and P. Traverso. 2004. *Automated Planning: Theory and Practice*. San Mateo, CA: Morgan Kaufmann Publishers.

Gilovich, T., D. Griffin, and D. Kahneman, eds. 2002. *Heuristics and Biases: The Psychology of Intuitive Judgment*. New York: Cambridge University Press.

Gilovich, T., R. Vallone, and A. Tversky. 1985. The hot hand in basketball: On the misperception of random sequences. *Cogn Psychol* 17:295–314.

Gong, L. 2008. The boundary of racial prejudice: Comparing preferences for computer-synthesized black, white and robot characters. *Comput Hum Behav* 24(5):2074–93.

Gordon, A. 2004. The representation of planning strategies. *Artif Intell* 153(1/2):287–305.

Gordon, S. E., and R. T. Gill. 1992. Knowledge acquisition with question probes and conceptual graph structures. In *Questions and Information Systems*, ed. T. Lauer, E. Peacock, and A. Graesser, 29–46. Hillsdale, NJ: Lawrence Erlbaum.

Gordon, S. E., and R. T. Gill. 1997. Cognitive task analysis. In *Naturalistic Decision Making*, ed. C. Zsambok and G. Klein, 131–40. Hillsdale, NJ: Lawrence Erlbaum.

Grabowski, M., and S. Sanborn. 2003. Human performance and embedded intelligent technology in safety-critical systems. *Int J Hum Comput Stud* 58(6):637–70.

Grier, R., J. Warm, and W. Dember. 2003. The vigilance decrement reflects limits in effortful attention, not mindlessness. *Hum Factors* 45(3):349–59.

Griffith, T. L., J. E. Sawyer, and M. A. Neale. 2003. Virtualness and knowledge in teams: Managing the love triangle of organizations, individuals and information technology. *MIS Q* 27:265–87.

Grondin, S. 2005. Overloading temporal memory. *J Exp Psychol Hum Percept Perform* 31(5):869–79.

Guerlain, S., P. J. Smith, J. H. Obradovich, S. Rudmann, P. Strohm, J. W. Smith, and J. Svirbely. 1996. Dealing with brittleness in the design of expert systems for immunohematology. *Immunohematology* 12:101–7.

Guerlain, S., P. J. Smith, J. H. Obradovich, S. Rudmann, P. Strohm, J. W. Smith, J. Svirbely, and L. Sachs. 1999. Interactive critiquing as a form of decision support: An empirical evaluation. *Hum Factors* 41:72–89.

Hammond, K. J. 1989. *Case-Based Planning: Viewing Planning as a Memory Task*. Boston, MA: Academic Press, Inc.

Handberg, R. 1995. Expert testimony on eyewitness identification—A new pair of glasses for the jury. *Am Crim Law Rev* 32(4): 1013–64.

Handy, C. 1995. Trust and the virtual organization. *Harv Bus Rev* 73:40–50.

Harris, S., and J. Owens. 1986. Some critical factors that limit the effectiveness of machine intelligence technology in military systems applications. *J Comput Based Instr* 13:30–34.

Haselton, M. G., D. Nettle, and P. W. Andrews. 2005. The evolution of cognitive bias. In *The handbook of evolutionary psychology*, ed. D. M. Buss, 724–46. Hoboken, NJ: John Wiley & Sons.

Hasling, D. W., W. J. Clancey, and G. Rennels. 1984. Strategic explanations for a diagnostic consultation system. *Int J Man Mach Syst* 20:3–19.

Hayes, C. C., and C. A. Miller, eds. 2010. Human computer etiquette: Should computers be polite? In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 1–12. Boca Raton, FL: Taylor & Francis.

Hayes, C. C., and U. Ravinder. 2003. Weasel: An automated planner that users can guide. In *Proceedings of the 2003 Conference on IEEE Systems, Man and Cybernetics*, October 5–8. Washington, DC. New York: IEEE.

Hayes-Roth, B., and F. Hayes-Roth. 1979. A cognitive model of planning. *Cogn Sci* 3(4):275–310.

Helander, M., T. Landauer, and P. Prabhu, eds. 1997. *Handbook of Human-Computer Interaction*. Amsterdam, Netherlands: Elsevier.

Hertel, G., S. Geister, and U. Konradt. 2005. Managing virtual teams: A review of current empirical research. *Hum Resour Manage Rev* 15(1):69–95.

Hester, R., and H. Garavan. 2005. Working memory and executive function: The influence of content and load on the control of attention. *Mem Cogn* 33(2):221–33.

Hinds, P., and S. Kiesler, eds. 2002. *Distributed Work*. Cambridge, MA: MIT Press.

Hinds, P. J., and M. Mortensen. 2005. Understanding conflict in geographically distributed teams: The moderating effects of shared identity, shared context, and spontaneous communication. *Organiz Sci* 16:290–307.

Hoc, J. -M. 2000. From human-machine interaction to human-machine cooperation. *Ergonomics* 43:833–43.

Hoc, J. M. 2001. Towards a cognitive approach to human-machine cooperation in dynamic situations. International Journal of Human-Computer Studies 54(4):509–540.

Hollnagel, E. 2003. *Handbook of Cognitive Task Design*. Mahwah, NJ: Lawrence Erlbaum.

Hollnagel, E., and D. Woods. 2005. *Joint Cognitive Systems: Foundations of Cognitive Systems Engineering*. Boca Raton, FL: Taylor & Francis.

Horton, F., and D. Lewis, eds. 1991. *Great Information Disasters*. London, UK: Association for Information Management.

Hutchins, E. 1990. The technology of team navigation. In *Intellectual Teamwork: Social and Technical Bases of Collaborative Work*, ed. J. Galegher, R. Kraut, and C. Egido. Hillsdale, NJ: Lawrence Erlbaum.

Hutchins, E. 1995. *Cognition in the Wild*. Cambridge, MA: MIT Press.

Jagacinski, R. J., and J. M. Flach. 2003. *Control Theory for Humans: Quantitative Approaches to Modeling Performance*. Hillsdale, NJ: Lawrence Erlbaum.

Janis, I. 1982. *Groupthink*, 2nd ed. Boston, MA: Houghton Mifflin.

Johnson, P., A. Duran, F. Hassebrock, J. Moller, M. Prietulla, P. Feltovich, and D. Swanson. 1981. Expertise and error in diagnostic reasoning. *Cogn Sci* 5:235–83.

Johnson, W. L., E. Shaw, A. Marshall, and C. LaBore. 2003. Evolution of User Interaction: the case of agent Adele. In *International Conference on Intelligent User Interfaces, Proceedings of the 8th International Conference on Intelligent User Interfaces*, 93–100. Miami, FL.

Johnson, W. L., and N. Wang. 2010. The Role of Politeness in Interactive Educational Software for Language Tutoring. In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 91–113. Boca Raton, FL: Taylor & Francis.

Johnston, W. A., and V. J. Dark. 1986. Selective attention. *Annu Rev Psychol* 37:43–75.

Jonassen, D., Tessmer, M. and Hannum, W. 1999. Task Analysis Methods for Instructional Design. Mahwah NJ: Lawrence Erlbaum.

Jonassen, D., M. Tessmer, and W. Hannum. 1999. *Task Analysis Methods for Instructional Design*. Mahwah, NJ: Lawrence Erlbaum.

Jones, P., and J. Jacobs. 2000. Cooperative problem solving in human-machine systems: Theory, models and intelligent associate systems. *IEEE Trans Syst Man Cybern* 30(4):397–407.

Jones, P. M., and C. M. Mitchell. 1995. Human-computer cooperative problem solving: Theory, design, and evaluation of an intelligent associate system. *IEEE Trans Syst Man Cybern* 25:1039–53.

Jones, E., and R. Nisbett. 1971. The actor and the observer: Divergent perceptions of the causes of behavior. In *Attributions: Perceiving the Causes of of Behavior*, ed. E. Jones, D. Kanouse, H. Kelley, R. Nisbett, S. Valins, and B. Weiner, 79–94. Morristown, NJ: General Learning Press.

Jones, D. R., and D. A. Schkade. 1995. Choosing and translating between problem representations. *Organiz Behav Hum Decis Process* 61(2):214–23.

Josephson, J., and S. Josephson. 1994. *Abductive Inference*, 31–8. New York: Cambridge University Press.

Kahneman, D., and A. Tversky, eds. 2000. *Choices, Values and Frames*. New York: Cambridge University Press.

Katz, N., D. Lazer, H. Arrow, and N. Contractor. 2004. Network theory and small groups. *Small Group Res* 35(3):307–32.

Kieras, D. 2004. GOMS models for task analysis. In *The Handbook of Task Analysis for Human-Computer Interaction*, ed. D. Diaper and N. Stanton, 83–116. Mahwah, NJ: Lawrence Erlbaum.

Kieras, D., and D. Meyer. 1997. An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Hum Comput Interact* 12:391–438.

Kirwan, B., and L. Ainsworth. 1992. *A Guide to Task Analysis*. London: Taylor & Francis.

Klayman, J., and Y. Ha. 1987. Confirmation, disconfirmation, and information in hypothesis testing. *Psychol Rev* 94:211–28.

Klein, G. A. 1993. A recognition-primed decision (RPD) model of rapid decision making. In *Decision Making in Action: Models and Method*, ed. G. Klein, J. Oransanu, R. Calderwood, and C. Zsambok, 138–47. Norwood, NJ: Ablex.

Klein, G. 2000. Cognitive task analysis of teams. In *Cognitive Task Analysis*, ed. J. M. Schraagen, S. Chipman, and V. Shalin, 417–30. Mahwah, NJ: Lawrence Erlbaum.

Klein, G., R. Caulderwood, and D. MacGregor. 1989. Critical decision method for eliciting knowledge. *IEEE Trans Syst Man Cybern* 19:462–72.

Kleinmuntz, D. N., and D. A. Schkade. 1993. Information displays and decision processes. *Psychol Sci* 4(4):221–7.

Koehler, J. J. 1996. The base rate fallacy reconsidered: Descriptive, normative, and methodological challenges. *Behav Brain Sci* 9(1)*:*1–53.

Kohler, W. 1992. *Gestalt Psychology: An Introduction to New Concepts in Modern Psychology*. New York: Liveright Publishing Corporation.

Kolodner, J. 1993. *Case-Based Reasoning*. San Mateo, CA: Morgan Kaufmann Publishers.

Kotovsky, K., J. R. Hayes, and H. A. Simon. 1985. Why are some problems hard? Evidence from Tower of Hanoi. *Cogn Psychol* 17:248–94.

Kulak, D., and E. Guiney. 2003. *Use Cases: Requirements in Context*. 2nd ed. Harlow, England: Addison-Wesley.

Kushleyeva, Y., D. Salvucci, F. Lee, and C. Schunn. 2005. Deciding when to switch tasks in time-critical multitasking. *Cogn Syst Res* 6(1):41–9.

Kutti, M. 1995. Activity theory as a potential framework for human-computer interaction research. In Nardi, B. (ed.) Context and consciousness: Activity theory and human-computer interaction. Cambridge MA: MIT Press.

Laird, J. E., A. Newell, and P. S. Rosenbloom. 1987. SOAR: An architecture for general intelligence. *Artif Intell* 33:1–64.

Larkin, J. H., and H. A. Simon. 1987. Why a diagram is (sometimes) worth ten thousand words. *Cogn Sci* 11:65–99.

Larson, A. and C. Hayes, 2005. An assessment of WEASEL: A Decision support system to assist in military planning. Proceedings of the 2005 Annual Meeting of the Human Factors Society Orlando, 287–91.

Larson, A. D. 2010. *The Impact of Computer Decision Support on Military Team Decision Making*. PhD Thesis, University of Minnesota, Minneapolis, MN, USA.

Larson, A. D., and C. C. Hayes. 2005. An assessment of WEASEL: A decision support system to assist in military planning. In *Proceedings of the 2005 Annual Meeting of the Human Factors and Ergonomics Society*, 287–91. Santa Monica, CA: Human Factors and Ergonomics Society.

Lee, J., and K. See. 2004. Trust in automation: Designing for appropriate reliance. *Hum Factors* 46(1):50–80.

Lehner, P. E., and D. A. Zirk. 1987. Cognitive factors in user/expert-system interaction. *Hum Factors* 29(1):97–109.

Levi, D. 2001. *Group Dynamics for Teams*. London, UK: SAGE Publications.

Lewis, C., and C. Wharton. 1997. Cognitive walkthroughs. In *Handbook of Human-Computer Interaction*, 2nd ed., ed. M. Helander, T. Landauer, and P. Prabhu, 717–31. Amsterdam, Netherlands: Elsevier.

Loftus, E. 1975. Leading questions and the eyewitness report. *Cogn Psychol* 7:560–72.

Logan, G. 2005. Attention, automaticity, and executive control. In *Experimental Cognitive Psychology and its Applications*, ed. A. F. Healy, 129–139. Washington, DC: American Psychological Association.

Lowe, R. K. 2003. Animation and learning: Selective processing of information in dynamic graphics. *Learn Instr* 13(2):157–76.

Luo, Y., P. Greenwood, and R. Parasuraman. 2001. Dynamics of the special scale of visual attention revealed by brain event-related potentials. *Cogn Brain Res* 12(3):371–81.

Lurey, J. S., and M. S. Raisinghani. 2001. An empirical study of best practices in virtual teams. *Inf Manage* 38:523–44.

Mack, R. 1995. Scenarios as engines of design. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 361–86. New York: John Wiley & Sons.

Mackworth, N. 1950. Researches on the measurement of human performance. Reprinted in *Selected papers on Human Factors in the Design and Use of Control Systems*, 1961, ed. H. W. Sinaiko. New York: Dover Publications.

Madani, O., A. Condon, and S. Hanks. 2003. On the undecidability of probabilistic planning and related stochastic optimization problems. *Artif Intell* 147(1/2):5–34.

Majercik, S., and M. Littman. 2003. Contingent planning under uncertainty via stochastic satisfiability. *Artif Intell* 147(1/2):119–62.

Marsh, S., and M. Dibben. 2003. *Annu Rev Inf Sci Technol* 37:465–98.

McCauley, C. 1989. The nature of social influence in groupthink: Compliance and internalization. *J Pers Soc Psychol* 22:250–60.

Meister, D., and T. Enderwick. 2002. *Human Factors in System Design, Development and Testing*. Mahwah, NJ: Lawrence Erlbaum.

Metzger, U., and R. Parasuraman. 2005. Automation in future air traffic management: Effects of decision aid reliability on controller performance and mental workload. *Hum Factors* 47(1):35–49.

Michalewicz, Z., and D. Fogel. 2004. *How to Solve It: Modern Heuristics*. New York: Springer.

Michie, D. 1986. *On Machine Intelligence*. 2nd ed. Chichester, West Sussex: Ellis Horwood Limited.

Mikulka, P., M. Scerbo, and F. Freeman. 2002. Effects of a biocybernetic system on vigilance performance. *Hum Factors* 44(4):654–64.

Miller, C., M. Pelican, and R. Goldman. 2000. "Tasking" interfaces to keep the operator in control. In *Proceedings of the 5th International Conference on Human Interaction with Complex Systems*. Urbana, IL.

Miller, G. A., E. Galanter, and K. H. Pribram. 1960. *Plans and the Structure of Behavior*. New York: Henry Holt and Company.

Miller, P. 1986. *Expert Critiquing Systems: Practice-Based Medical Consultation by Computer*. New York: Springer-Verlag.

Mital, A., and A. Pennathur. 2004. Advanced technologies and humans in manufacturing workplaces: An interdependent relationship. *Int J Ind Ergon* 33(4):295–313.

Mitchell, C. M., and R. A. Miller. 1986. A discrete control model of operator function: A methodology for information display design. *IEEE Trans Syst Man Cybern* 16:343–57.

Muir, B. 1987. Trust between humans and machines. *Int J Man Mach Stud* 27:527–39.

Muir, B., and N. Moray. 1996. Trust in automation 2: Experimental studies of trust and human intervention in a process control simulation. *Ergonomics* 39(3):429–60.

Mumford, M., R. Schultz, and J. Van Doorn. 2001. Performance in planning: Processes, requirement, and errors. *Rev Gen Psychol* 5(3):213–40.

Myers, D. 2004. *Psychology*. 7th ed. New York: Worth Publishers.

Mynatt 1977: Mynatt, C., Doherty, M. and Tweney, R. 1977. Confirmation bias in a simulated research environment: An experimental study of scientific inference. Quarterly Journal of Experimental Psychology 30:85–95.

Mynatt, C., M. Doherty, and R. Tweney. 1977. Confirmation bias in a simulated research environment: An experimental study of scientific inference. *Q J Exp Psychol* 30:85–95.

Nakatsu, R. T., and I. Benbasat. 2003. Improving the explanatory power of knowledge-based systems: An investigation of content and interface-based enhancements. *IEEE Trans Syst Man Cybern Part A Syst Hum* 33(3):344–57.

Nareyek, A., R. Fourer, and E. Freuder. 2005. Constraints and AI planning. *IEEE Intell Syst* 20(2):62–72.

Nason, S., J. Laird, and C. Schunn. 2005. SOAR-RL: Integrating reinforcement learning with SOAR. *Cogn Syst Res* 6(1):51–9.

Nass, C., J. Steuer, and E. Tauber. 1994. Computers are social actors. In *CHI '94, Human Factors in Computing Systems*. New York: Association for Computing Machinery.

Nass, C., Y. Moon, and N. Green. 1997. Are Machines Gender-Neutral? Gender Stereotypic Responses to Computers. *J Appl Psychol* 27(10):864–76.

Navon, D. 1978. The importance of being conservative. British Journal of Mathematical and Statistical Psychology 31:33–48.

Newell, A. 1990. *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Newell, A., and H. Simon. 1972. *Human Problem Solving*. Englewood Cliffs, NJ: Prentice Hall, Inc.

Nickerson, R. 1988. Counting, computing, and the representation of numbers. *Hum Factors* 30:181–99.

Norman, D. A. 1981. Categorization of action slips. *Psychol Rev* 88(1):1–15.

Norman, D. A. 2002. *The Design of Everyday Things*. New York: Doubleday.

Oliver, D., and J. Roos. 2005. Decision making in high-velocity environments. *Organiz Stud* 26(6):889–913.

Olson, G. M., and J. S. Olson. 1997. Research on computer supported cooperative work. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. Landauer, and P. Prabhu, 1433–56. Amsterdam, Netherlands: Elsevier.

Olson, G., and J. Olson. 2003. Groupware and computer-supported cooperative work. In *Handbook of Human-Computer Interaction*, ed. A. Sears and J. Jacko, 583–93. Mahwah, NJ: Lawrence Erlbaum.

Orasanu, J., and E. Salas. 1993. Team decision making in complex environments. In *Decision Making in Action: Models and Methods*, ed. G. Klein, R., J. Orusanu, and Calderwood, 327–45. New Jersey: Ablex.

Ormrod, J. 2003. *Human Learning*. 4th ed. Englewood Cliffs, NJ: Prentice Hall.

Parasuraman, R. 2000. Designing automation for human use: Empirical studies and quantitative models. *Ergonomics* 43:931–51.

Parasuraman, R., and V. Riley. 1997. Humans and automation: Use, misuse, disuse and abuse. *Hum Factors* 39:230–53.

Parasuraman, R., T. B. Sheridan, and C. D. Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Trans Syst Man Cybern* 30(3):286–97.

Pashler, H., J. C. Johnston, and E. Ruthruff. 2001. Attention and performance. *Annu Rev Psychol* 52:629–51.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufman.

Pennington, N., and R. Hastie. 1992. Explaining the evidence: Tests of the story model for juror decision making. *J Pers Soc Psychol* 62:189–206.

Petroski, H. 1994. *Design Paradigms: Case Histories of Error and Judgment in Engineering*. New York: Cambridge University Press.

Plous, S. 1993. *The Psychology of Judgment and Decision Making*. New York: McGraw-Hill.

Poulton, E. C. 1989. *Bias in Quantifying Judgments*. Hillsdale, NJ: Lawrence Erlbaum.

Preece, J., Y. Rogers, and H. Sharp. 2002. *Interaction Design*. New York: John Wiley & Sons.

Preece, J., Y. Rogers, H. Sharp, and D. Benyon. 1994. *Human-Computer Interaction*. Reading, MA: Addison-Wesley.

Pritchett, A., B. Vandor, and K. Edwards. 2002. Testing and implementing cockpit alerting systems. *Reliab Eng Syst Saf* 75(2):193–206.

Psillos, S. 2002. Simply the best: A case for abduction. *Lect Notes Artif Intell* 2408:605–25.

Radvansky, G. 2005. *Human Memory*. Boston, MA: Allyn and Bacon.

Rasmussen, J. 1983. Skills, rules and knowledge: Signals, signs, symbols and other distinctions in human performance models. *IEEE Trans Syst Man Cybern* SMC-13(3):257–66.

Rasmussen, J., B. Brehner, and J. Leplat, eds. 1991. *Distributed Decision Making: Cognitive Models for Cooperative Work*. New York: John Wiley & Sons.

Rasmussen, J., A. Pejtersen, and L. Goldstein. 1994. *Cognitive Systems Engineering*. New York: John Wiley & Sons.

Ravinder, U. 2003. *Weasel: A Constraint-Based Tool for Generating Enemy Courses of Action*. Masters Thesis, University of Minnesota.

Rayward-Smith, V., I. Osman, C. Reeves, and G. Smith, eds. 1996. *Modern Heuristic Search Methods*. New York: John Wiley & Sons.

Reason, J. 1991. *Human Error*. New York: Cambridge Press.

Reason, J. 1997. *Managing the Risks of Organizational Accidents*. Hampshire, UK: Ashgate.

Reason, J. T. 1990. Human Error. New York: Cambridge University Press.

Riegelsberger, J., M. Sasse, and J. McCarthy. 2005. The mechanics of trust: A framework for research and design. *Int J Hum Comput Stud* 62(3):381–422.

Reising, D., and P. Sanderson. 2002. Work domain analysis and sensors I: Principles and simple example. *Int J Hum Comput Stud* 56(6):569–96.

Riesbeck, C. K., and R. C. Schank. 1989. *Inside Case–Based Reasoning*. Hillsdale, NJ: Lawrence Erlbaum.

Roth, E. M., K. B. Bennett, and D. D. Woods. 1987. Human interaction with an 'intelligent' machine. *Int J Man Mach Stud* 27:479–525.

Rouse, W. B. 1980. *Systems Engineering Models of Human Machine Interaction*. New York: Elsevier.

Rubin, K. S., P. M. Jones, and C. M. Mitchell. 1988. OFMSpert: Interfence of operator intentions in supervisory control using a blackboard structure. *IEEE Trans Syst Man Cybern* 18(4):618–37.

Russell, S., and P. Norvig. 1995. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Sacerdoti, E. D. 1974. Planning in a hierarchy of abstraction spaces. *Artif Intell* 5(2):115–35.

Salas, E., C. Bowers, and E. Edens, eds. 2001. *Improving Teamwork in Organizations: Applications of Resource Management Training*. Mahwah, NJ: Lawrence Erlbaum.

Salas, E., and S. Fiore, eds. 2004. *Team Cognition: Understanding the Factors that Drive Process and Performance*. Washington, DC: American Psychological Association.

Sarter, N., and D. Woods. 1993. *Cognitive Engineering in Aerospace Applications: Pilot Interaction with Cockpit Automation*. NASA Contractor Report 177617, NASA Ames Research Center, Moffett Field, CA.

Schaeken, W., ed. 2005. *Mental Models Theory of Reasoning: Refinements and Extensions*. Mahwah, NJ: Lawrence Erlbaum.

Schraagen, J. M., Chipman, S. and Shalin, V. (eds.). 2000. Cognitive Task Analysis. Mahwah NJ: Lawrence Erlbaum.

Schraagen, J. M., S. Chipman, and V. Shalin. eds. 2000. *Cognitive Task Analysis*. Mahwah, NJ: Lawrence Erlbaum.

Schroeder, R., and A. -S. Axelsson, eds. 2005. *Work and Plan in Shared Virtual Environments: Computer Supported Cooperative Work*. New York: Springer.

Schuler, D., and A. Namioka, eds. 1993. *Participatory Design: Principles and Practices*. Mahwah, NJ: Lawrence Erlbaum.

Scaife, M., and Y. Rogers. 1996. External cognition: How do graphical representations work? *Int J Hum Comput Stud* 45:185–213.

Scott, R., E. M. Roth, and S. E. Deutsch. 2005. Work-centered support systems: A human-centered approach to intelligent system design. *IEEE Intell Syst* 20:73–81.

Seamon, J. 2005. *Human Memory: Contemporary Readings*. Cambridge: Oxford University Press.

Sewell, D. R., and N. D. Geddes. 1990. A plan and goal based method for computer-human system design. In *Human computer Interaction: INTERACT 90*, 283–8. New York.

Shafer, G. 1996. *Probabilistic Expert Systems*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Shafer, G., and J. Pearl, eds. 1990. *Readings in Uncertain Reasoning*. San Mateo, CA: Morgan Kaufmann.

Shalin, V. L., N. D. Geddes, D. Bertram, M. A. Szczepkowski, and D. DuBois. 1997. Expertise in dynamic physical task domains. In *Expertise in Context: Human and Machine*, ed. P. Feltovich, K. Ford, and R. Hoffman, 194–217. Cambridge, MA: MIT Press.

Shepherd, A. 2000. *Hierarchical Task Analysis*. London: Taylor & Francis.

Sheridan, T. B. 1997. Supervisory control. In *Handbook of Human Factors*, 2nd ed., ed. G. Salvendy, 1295–327. New York: John Wiley & Sons.

Sheridan, T. 2002. *Humans and Automation: System Design and Research Issues*. Chidester, England: Wiley.

Sheridan, T. B., and W. R. Ferrell. 1974. *Man–Machine Studies: Information, Control, and Decision Models of Human Performance*. Cambridge, MA: MIT Press.

Shortliffe, E. 1990. Clinical decision support systems. In *Medical Informatics: Computer Applications in Health Care*, ed. E. Shortliffe and L. Perreault, 466–500. New York: Addison-Wesley.

Silverman, B. G. 1992. Survey of expert critiquing systems: Practical and theoretical frontiers. *Commun ACM* 35(4):106–28.

Simmons, R. G. 1988. A Theory of debugging plans and Interpretations. In *Proceedings of the American Association for Artificial Intelligence (AAAI-98)*, 94–9. Minneapolis, MN. Menlo Park, CA: American Association for Artificial Intelligence Press.

Skitka, L., K. Mosier, and M. Burdick. 1999. Does automation bias decision making? *Int J Hum Comput Syst* 51:991–1006.

Skitka, L., Mosier, K. and Burdick, M. 1999. Does automation bias decision making? International Journal of Human-Computer Systems 51:991–1006.

Smith, P. J., R. Beatty, A. Spencer, and C. Billings. 2003. Dealing with the challenges of distributed planning in a stochastic environment: Coordinated contingency planning. In *Proceedings of the 2003 Annual Conference on Digital Avionics Systems*. Reston, VA: American Institute of Aeronautics and Astronautics.

Smith, P. J., C. Billings, R. Chapman, J. H. Obradovich, E. McCoy, and J. Orasanu. 2000. Alternative architectures for distributed cooperative problem-solving in the national airspace system. In *Proceedings of the 5th International Conference on Human Interaction with Complex Systems*, ed. M. Benedict. Urbana, IL.

Smith, P. J., and N. Geddes. 2003. A cognitive systems engineering approach to the design of decision support systems. In *Handbook of Human-Computer Interaction*, ed. A. Sears and J. Jacko, 656–75. Mahwah, NJ: Lawrence Erlbaum.

Smith, P. J., W. Giffin, T. Rockwell, and M. Thomas. 1986. Modeling fault diagnosis as the activation and use of a frame system. *Hum Factors* 28(6):703–16.

Smith, P. J., E. McCoy, and C. Layton. 1997. Brittleness in the design of cooperative problem-solving systems: The effects on user performance. *IEEE Trans Syst Mana Cybern* 27(3):360–71.

Smith, P. J., E. McCoy, and J. Orasanu. 2000. Distributed cooperative problem-solving in the air traffic management system. In *Naturalistic Decision Making*, ed. G. Klein and E. Salas, 369–84. Mahwah, NJ: Lawrence Erlbaum.

Smith, P. J., E. McCoy, and J. Orasanu. 2001. Distributed cooperative problem-solving in the air traffic management system. In *Naturalistic Decision Making*, ed. G. Klein and E. Salas, 369–84. Mahwah, NJ: Lawrence Erlbaum.

Smith, P. J., E. McCoy, J. Orasanu, C. Billings, R. Denning, M. Rodvold, T. Gee, and A. VanHorn. 1997. Control by permission: A case study of cooperative problem-solving in the interactions of airline dispatchers and ATCSCC. *Air Traffic Control Q* 4:229–47.

Smith, P. J., J. H. Obradovich, S. Guerlain, S. Rudmann, P. Strohm, J. Smith, J. Svirbely, and L. Sachs. 1998. Successful use of an expert system to teach diagnostic reasoning for antibody identification. In *Proceeding of the 4th International Conference on Intelligent Tutoring Systems*, 354–63. San Antonio, Texas. Berlin: Springer.

Smith, P. J., and S. Rudmann. 2005. Clinical decision making and diagnosis: Implications for immunohematologic problem-solving. In *Serologic Problem-Solving: A Systematic Approach for Improved Practice*, ed. S. Rudmann, 1–16. Bethesda, MD: AABB Press.

Smith, P. J., R. B. Stone, and A. Spencer. 2006. Design as a prediction task: Applying cognitive psychology to system development. In *Handbook of Industrial Ergonomics*, 2nd ed., ed. W. Marras and W. Karwowski. New York: Marcel Dekker, Inc.

St. Amant, R., A. Freed, F. Ritter, and C. Schunn. 2005. Specifying ACT-R models of user interaction with a GOMS language. *Cogn Syst Res* 6(1):71–88.

Sternberg, R. J., and J. Pretz. 2005. *Cognition and Intelligence: Identifying the Mechanisms of the Mind*. New York: Cambridge University Press.

Strauch, B. 2004. *Investigating Human Error: Incidents, Accidents, and Complex Systems*. Hampshire, UK: Ashgate Publishing.

Suchman, L. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication*. New York: Cambridge University Press.

Sycara, K., and M. Lewis. 2004. Integrating intelligent agents into human teams. In *Team Cognition: Understanding the Factors that Drive Process and Performance*, ed. E. Salas and S. Fiore, 203–31. Washington, DC: American Psychological Association.

Tetlock, P. 1998. Social psychology and world politics. In *The Handbook of Social Psychology*, 4th ed., ed. D. Gilbert, S. Fiske, and G. Lindzey, Vol. 2, 868–912. New York: McGraw-Hill.

Tetlock, P., R. Peterson, M. McQuire, S. Chang, and P. Feld. 1992. Assessing political group dynamics: A test of the groupthink model. *J Personal Soc Psychol* 63:403–25.

Tianfield, H., and R. W. Wang. 2004. Critic systems—Towards human-computer collaborative problem solving. *Artif Intell Rev* 22(4):271–95.

Tobey, D. 2005. *Needs Assessment Basics*. Alexandria, VA: ASTD Press.

Tufte, E. R. 1983. *The Visual Display of Quantitative Information*. Chesire, CT: Graphics Press.

Tufte, E. R. 1990. *Envisioning Information*. Chesire, CT: Graphics Press.

Tufte, E. R. 1997. *Visual Explanations*. Cheshire, CT: Graphics Press.

Turban, E., J. Aronson, and T. -P. Liang. 2004. *Decision Support Systems and Intelligent Systems*. 7th ed. Englewood Cliffs, NJ: Prentice Hall.

Tversky, A. 1972. Elimination by aspects: A theory of choice. *Psychol Rev* 79:281–99.

Tversky, A. 1982. *Judgment Under Uncertainty: Heuristics and Biases*. New York: Cambridge University Press.

Tversky, A., and D. Kahneman. 1974. Judgment under uncertainty: Heuristics and biases. *Science* 185:1124–31.

Vicente, K. J. 1991. *Supporting Knowledge-Based Behavior Through Ecological Interface Design* (EPRL-91-02). Urbana-Champaign, IL: Engineering Psychology Research Laboratory, Department of Mechanical Engineering, University of Illinois.

Vicente, K. 1999. *Cognitive Work Analysis: Toward Safe, Productive, and Healthy Computer-Based Work*. Mahwah, NJ: Lawrence Erlbaum.

Vicente, K. 2000. Work domain analysis and task analysis: A difference that matters. In *Cognitive Task Analysis*, ed. J. M. Schraagen, S. Chipman, and V. Shalin, 101–18. Mahwah, NJ: Lawrence Erlbaum.

Vicente, K. J. 2002. Ecological interface design: Progress and challenges. *Hum Factors* 44(1):62–78.

Vicente, K., and J. Rasmussen. 1990. The ecology of human-machine systems II: Mediating "direct perception" in complex work domains. *Ecol Psychol* 2:207–49.

Vicente, K., and J. Rasmussen. 1992. Ecological interface design: Theoretical foundations. *IEEE Trans Syst Man Cybern* 22:589–606.

Wang, N., W. L. Johnson, P. Rizzo, E. Shaw, and R. Mayer. 2005. Experimental evaluation of polite interaction tactics for pedagogical agents. In *Proceedings of the 10th International Conference on Intelligent User Interfaces*, 12–19. San Diego, CA.

Wason, P. 1960. On the failure to eliminate hypotheses in a conceptual task. *Q J Exp Psychol* 12:129–40.

Watzman, S. 2003. Visual design principles for usable interfaces. In *Handbook of Human-Computer Interaction*, ed. A. Sears and J. Jacko, 263–85. Mahwah, NJ: Lawrence Erlbaum.

Wickens, C. D., J. Lee, Y. Liu, and S. G. Becker. 2004. *An Introduction to Human Factors Engineering*. 2nd ed. Upper Saddle River, NJ: Pearson/Prentice Hall.

Wiegmann, D., and S. Shappell. 2003. *A Human Error Approach to Aviation Accident Analysis: The Human Factors Analysis and Classification System*. Hampshire, UK: Ashgate.

Wiegmann, D., H. Zhang, T. Von Thaden, G. Sharma, and A. Gibbons. 2004. Safety culture: An integrative review. *Int J Aviat Psychol* 14(2):117–34.

Wiener, E. L., and D. C. Nagel. 1988. *Human Factors in Aviation*. New York: Academic Press.

Wilensky, R. (1983). Planning and Understanding: A Computational Approach to Human Reasoning. Reading MA: AddisonWesley.

Williams, K. 2005. Computer-aided GOMS: A description and evaluation of a tool that integrates existing research for modeling human-computer interaction. *Int J Hum Comput Interact* 1(1):39–58.

Witkin, B. R., and J. S. Altshuld. 1995. *Planning and Conducting Needs Assessments*. Thousand Oaks, CA: Sage Publications.

Wogalter, M. S., and C. B. Mayhorn. 2005. Providing cognitive support with technology-based warning systems. *Ergonomics* 48(5):522–33.

Wu, P., C. Miller, H. Funk, and V. Vikili. 2010. Computational Models of Etiquette and Culture. In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 63–89. Boca Raton, FL: Taylor & Francis.

Yantis, S. 1998. Control of visual attention. In *Attention*, ed. H. Pashler, 223–56. Hillsdale, NJ: Lawrence Erlbaum.

Zhang, J. 1997. The nature of external representations in problem solving. *Cogn Sci* 21(2):179–217.

Zhang, J., and D. A. Norman. 1994. Representations in distributed cognitive tasks. *Cogn Sci* 18:87–122.

Zhang, T., B. Zhu, and D. B. Kaber. 2010. Anthropomorphism and Social Robots: Setting Etiquette Expectations. In *Human-Computer Etiquette: Cultural Expectations and the Design Implications They Place on Computers and Technology*, ed. C. Hayes and C. Miller, 231–59. Boca Raton, FL: Taylor & Francis.

Zuboff, S. 1988. *In the Age of Smart Machines*. New York: Basic Books.

27 Online Communities

Panayiotis Zaphiris, Chee Siang Ang, and Andrew Laghos

CONTENTS

27.1 INTRODUCTION

It is estimated that there were over 1.70 billion Internet users globally in 2009 (Pingdom 2010). The expansion of the Internet has resulted in an increase in the usefulness of computer-mediated communication (CMC) and the popularity of online communities. One in four Internet users has participated in chat rooms or online discussions (Madden and Rainie 2003). Just on the social network site Facebook, there are around 350 million users, and 50% of these users log in everyday (Pingdom 2010).

This chapter begins by defining online communities and CMC and by discussing their main pros and cons. This is followed by a review of the different types of CMC, while the evolution of game-based and wiki communities are described in more depth. These two relatively new areas of online sociability create new opportunities and challenges in the way people work, learn, and play online.

Massively multiplayer online role-playing games (MMORPGs) have taken the social aspects of computer game playing to a new dimension, where players interact, socialize, and form networks of communities by having fun online. Wiki-based communities facilitate new models of social collaboration, where the creation of online content is no longer an individual action but rather it is transformed into a social, collaborative activity.

The analysis and evaluation of online communities require a good understanding of all the available evaluation frameworks and methodologies that exist. Online communities are a source of valuable data that, when properly analyzed, can provide us with insights about the social experience people who are part of them have. For this reason, we provide a description of the key methods in Section 27.3. We then demonstrate the application of some of these methods to characteristic case studies. Our first case study looks at how learning communities can be analyzed and how results from this analysis can be used for improving the pedagogical value of e-learning. The second case study investigates the use of activity theoretical frameworks in the analysis of computer game-based communities.

The chapter concludes with a brief summary and suggestions for new directions in the area of online communities.

27.2 COMPUTER-MEDIATED COMMUNICATION AND ONLINE COMMUNITIES

27.2.1 Computer-Mediated Communication

One of the most important characteristics of this medium is the opportunities it offers for human–human communication through computer networks. As Metcalfe (1992) points out, communication is the Internet's most important asset. E-mail is just one of the many modes of communication that can occur through the use of computers. Jones (1995) points out that through communication services, such as the Internet, Usenet, and bulletin boards, online communication has for many people supplanted the postal service, telephone, and even the fax machine. All these applications where the computer is used to mediate communication are called CMC.

December (1997) defines CMC as "the process by which people create, exchange, and perceive information using networked telecommunication systems (or nonnetworked computers) that facilitate encoding, transmitting, and decoding messages." He emphasizes that studies of CMC view this process from different interdisciplinary theoretical perspectives (social, cognitive/psychological, linguistic, cultural, technical, and political) and often draw from fields such diverse as human communication, rhetoric and composition, media studies, human–computer interaction (HCI), journalism, telecommunications, computer science, technical communication, and information studies.

27.2.2 Online Communities

Online communities emerge through the use of CMC applications. The term online community is multidisciplinary in nature, meaning different things to different people, and is slippery to define (Preece 2000). For purposes of a general understanding of what online communities are, Rheingold's definition of online communities is presented:

> [online] communities are social aggregations that emerge from the Net when enough people carry on those public discussions long enough, with sufficient human feeling, to form webs of personal relationships in cyberspace. (Rheingold 1993, p. 5)

Online communities are also often referred to as cyber societies, cyber communities, web groups, virtual communities, web communities, virtual social networks, and e-communities among several others.

The cyberspace is the new frontier in social relationships, and people are using the Internet to make friends, colleagues, lovers, and enemies (Suler 2004). As Korzenny (1978) pointed out, even as early as 1978, online communities are formed around interests and not physical proximity. In general, what brings people together in an online community is common interests such as hobbies, ethnicity, education, and beliefs. As Wallace (1999) points out, meeting in online communities eliminates prejudging based on someone's appearance, and thus people with similar attitudes and ideas are attracted to each other.

The emergence of the so called "global village" was predicted years ago (McLuhan 1964) as a result of television and satellite technologies. However, it is argued by Fortner (1993) that "global metropolis" is a more representative term (Choi and Danowski 2002). If one takes into account that the estimated world population of 2002 was 6.2 billion (U.S. Census Bureau 2004), then the online population is nearly 10% of the world population—a significant percentage that must be taken into account when analyzing online communities. In most online communities, time, distance, and availability are no longer disseminating factors. Given that the same individual may be part of several different and numerous online communities, it is obvious why online communities keep increasing in numbers, size, and popularity.

CMC has its benefits and limitations. For instance, CMC discussions are often potentially richer than face-to-face discussions. However, users with poor writing skills may be at a disadvantage when using text-based CMC (SCOTCIT 2003).

27.2.3 Examples of Computer-Mediated Communication and Online Communities

Examples of CMC include asynchronous communication like e-mail and bulletin boards; synchronous communication like chatting; and information manipulation, retrieval, and storage through computers and electronic databases (Ferris 1997).

Audio conferencing is a real-time communication mechanism, because the communication happens synchronously. Depending on the application, text chat and graphics may also be supported. Videoconferencing, like audio conferencing, offers a useful mode of communication, but has the added benefit of also being able to see the participants, instead of just hearing them.

Internet relay chats (IRCs) and chats also support synchronous communication, since they enable the users to carry out conversations through the use of text messaging. Multi-user domains (MUDs) build on chats by providing avatars and graphical environments where the users can engage in interactive fantasy games (Preece 2000).

www websites are usually asynchronous, providing community information and links to other sites, but sometimes also have synchronous software, like chats, embedded in them (Preece 2000).

E-mail is an asynchronous mode of communication usually in the form of text. However, the ability to add attachments to e-mail messages makes it possible for audio, video, and graphics to be used also. Voice mail is an expansion of e-mail whereby users may record themselves speaking out a message and then send that voice file to their contact, instead of typing it. Newsgroups, like e-mail, provide an asynchronous mode of communication, but unlike e-mail where the messages come to the users, it is a "pull" technology, meaning the users must go to the UseNet groups themselves (Preece 2000). Finally, discussion boards, also referred to as forums or bulletin boards, provide an asynchronous mode of communication where the users post messages for others to see and respond at their own time.

In Section 27.2.3.1, we describe a special type of online communities, that of online virtual game communities.

27.2.3.1 Online Virtual Game Communities

With the advent of ubiquitous broadband Internet connection and the increasing graphical processing power of personal computers, a new paradigm of gaming has emerged. MMORPGs have changed the game industry dramatically. MMORPGs provide a fictional setting where a large group of users voluntarily immerse themselves in a graphical virtual environment and interact with each other by forming a community of users.

Although the concept of multiplayer gaming is not new, the game world of most local network multiplayer games, as opposed to MMORPG, are simplistic and can accommodate only around 16 concurrent players in a limited space.

A MMORPG enables thousands of players to play in an evolving virtual world simultaneously over the Internet. The game world is usually modeled with highly detailed three-dimensional (3D) graphics, allowing individuals to interact not only with the gaming environment but also with other players. Usually, this involves the players representing themselves through the use of avatars—the visual representation of the player's identity in the virtual world.

The MMORPG environment is a new paradigm in computer gaming in which players are part of a persistent world, a world that exists independent of the users (Yee 2005). Unlike other games where the virtual world cease to exist when players switch off the game, in an MMORPG, the world exists before the user logs on and continues to exist when the user logs off. More importantly, events and interactions occur in the world even when the user is not logged on, as there are many other players who are constantly interacting, thus transforming the world. To accommodate the large number of users, the worlds in MMORPGs are vast and varied in terms of "geographical locations," characters, monsters, items, and so on. More often than not, new "locations" or items are added by the game developers from time to time according to the demand of the players.

On the one hand, an MMORPG, like a role-playing game (RPG), involves killing monsters, collecting items, developing characters, and so on. However, it contains an extra aspect, which is the internal sociability within the game. Unlike single player games, which rely on other external modes of communication (such as mailing lists, discussion forums outside the game) to form the gaming culture, the culture is formed within the MMORPG environment itself.

In such a way, these MMORPG virtual worlds represent the persistent social and material world, which is structured around narrative themes (usually fantasy), where players are engaged in various activities: they slay monsters, attack castles, scavenge goods, trade merchandise, and so on. On the one hand, the game's virtual world represent the escapist fantasy; on the other hand, it supports social realism (Kolbert 2001).

That means games are no longer meant to be a solitary activity played by a single individual. Instead, the player is expected to join a virtual community that is parallel with the physical world, in which societal, cultural, and economical systems arise. It has been gradually becoming a world that allows players to immerse into experiences that closely match those of the real world: virtual relationship is sought, virtual marriage is held, virtual shops are set up, and so on.

The MMORPG genre now boasts hundreds of thousands of users and accounts for millions of dollars in revenue each year. The number of people who play the games (and the time they invest in terms of activities within and around the game) is astounding. The MMORPG, Lineage (NCsoft 2005), for example, had more than 2.5 million current subscribers in 2002 (Vaknin 2002) and, within a year, Ultima Online (Electronic Art 2005) attracted more than 160 million person-hours (Kolbert 2001).

Such games are ripe for cultural analysis of the social practices around them. Although fundamentally, MMORPGs are video games with virtual spaces with which the players interact, they should be regarded not just as a piece of game software; they are a community, a society, and if you wish, a culture. These games are becoming the most interesting interactive CMC and networked activity environment (Taylor 2002). Thus, understanding the pattern of participation in these game communities is crucial, as such virtual communities function as a major mechanism of socialization of the players to the norms of the game culture that emerges, as Squire and Steinkeuhler (in press) has noted:

> Playing one's character(s) and living in [these virtual worlds] becomes an important part of daily life. Since much of the excitement of the game depends on having personal relationships and being part of [the] community's developing politics and projects, it is hard to participate just a little (Squire and Steinkeuhler in press).

Recently, game designers have tried to stretch the boundaries further by structuring in-game activities to maximize interaction. One of the examples of sociability by design in MMORPG is Star Wars Galaxies (Sony 2005), which is organized so that players are steered toward certain locations in the game world where social playing is expected to take place (Ducheneaut, Moore, and Nickell 2004).

Such communities formed around the game can be broadly divided into two categories: in-game and out-of-game communities. Most MMORPGs are created to encourage long-term relationships among the players through the features that support the formation of in-game communities. One of the most evident examples is the concept of guilds. Guilds are a fundamental component of MMORPG culture for people who are natural organizers to run a virtual association that has formalized membership and rank assignments to encourage participation. Sometimes, a player might join a guild and get involved in a guild war to fight for the castle. Each guild usually has a leader and several guilds could team up in a war. This involves a complicated leader–subordinate and leader–leader relationship.

In addition, to encourage social interaction, MMORPGs are specially designed in such a way that some game goals are almost impossible to be achieved without forming communities. For example, one player alone could spend a long period of time collecting all the items needed to assemble a device. But a guild could ask its members to fan out in small groups and collect all the necessary components in one day. Complex devices beyond the reach of any individual player could be quickly constructed by the guild. The guild could also accept donations from members and then distribute those contributions to others according to their needs, benefiting everyone as a result of this collaboration (Kelly 2004).

Apart from relatively long-term relationships such as guild communities, MMORPGs also provide many opportunities for short-term relationship experiences. For example, a player could team-up with another player to kill monsters to develop the abilities of their avatars (level up) or some more expert players could help newer players to get through the game.

When trying to win the game, players often need to get information from other resources: guidebooks, discussion forums, other players, and so on. Therefore, game playing is generally more concerned with player–player interaction than with player–game interaction. What is at first confined to the game alone soon spills over into the virtual world beyond it (e.g., websites, chat rooms, and e-mail) and even life off-screen (e.g., telephone calls, face-to-face meetings).

Apart from these external communities around the game that are mediated through e-mails or online forums (which also exist in many other games), there is an interesting phenomenon that fuses the internal and external game communities. The participation in an external community starts to break the magic circle of the game—that game space is no longer separate from real life—as the out-of-game community trades in-game items for real money.

For example, Norrath, the world of EverQuest, was estimated to have the seventy-seventh largest economy in the real world based on buying and selling in online auction houses.

> About 12,000 people call it their permanent home, although some 60,000 are present there at any given time. The nominal hourly wage is about USD3.42 per hour, and the labors of the people produce a GNP per capita somewhere between that of Russia and Bulgaria. A unit of Norrath's currency is traded on exchange markets at USD0.0107, higher than the Yen and the Lira. (Castronova 2001, p. 1)

Having illustrated the social phenomenon around such playful virtual community, it is believed that it is fruitful to research such communities as we might be able to derive some useful implications on how successful computer-supported collaborative work and computer-supported collaborative learning environments can be designed. For this reason, in Section 27.3, we will describe some of the methodologies that can be used in such studies, and in Section 27.4, we will present the application of some of these methods to two case studies.

27.3 ANALYZING ONLINE COMMUNITIES: FRAMEWORKS AND METHODOLOGIES

There are various aspects and attributes of CMC that can be studied to help us better understand online communities: for instance, the analysis of the frequency of exchanged messages and the formation of social networks or the analysis of the content of the exchanged messages and the formation of virtual communities. To achieve such an analysis, a number of theoretical frameworks have been developed and proposed. For example, Henri (1992) provides an analytical model for cognitive skills that can be used to analyze the process of learning within messages exchanged between students of various online e-learning communities. Mason's work (1991) provides descriptive methodologies using both quantitative and qualitative analysis. Furthermore, five phases of interaction analysis are identified in Gunawardena, Lowe, and Anderson model (1997):

1. Sharing/comparing of information
2. The discovery and exploration of dissonance or inconsistency among ideas, concepts, or statements
3. Negotiation of meaning/co-construction of knowledge
4. Testing and modification of proposed synthesis or co-construction
5. Agreement statement(s)/applications of newly constructed meaning

In this section, we provide a description of some of the most commonly used online community evaluation techniques and their weaknesses and strengths.

27.3.1 Query-Based Techniques and User Profiles: Interviews, Questionnaires, and Personas

27.3.1.1 Interviews

An interview can be defined as a type of conversation that is initiated by the interviewer to obtain research relevant information. The interview reports have to be carefully targeted and analyzed to make an impact (Usability Net 2003). Interviews are usually carried out on a one-to-one basis, where the interviewer collects information from the interviewee. Interviews can take place by telephone and face-to-face (Burge and Roberts 1993). They can also take place via nonreal-time methods like fax and e-mail, although in these cases, they function like questionnaires. Interviews are useful for obtaining information that are difficult to elicit through approaches such as background knowledge and general principles. There are three types of interviews: (1) structured interviews: consist of predetermined questions asked in fixed order, like a questionnaire; (2) semistructured interviews: questions are determined in advance but may be reordered, reworded, omitted, and elaborated; (3) unstructured interviews: it is not based on predetermined questions but instead the interview has a general area of interest and the conversation may develop freely.

Interviews can be used to gain insights about general characteristics of the participants of an online community and their motivation for participating in the community under investigation. The data collected comes straight from the participants of the online communities, whereby they are able to provide feedback based on their own personal experiences, activities, thoughts, and suggestions.

Advantages of interviews (Usability Net 2003) include the following: what is talked about can address directly the informant's individual concerns; mistakes and misunderstandings can be quickly identified and cleared up; more flexible than a questionnaire; can cover low-probability events.

Disadvantages of interviews include the following: danger of analyst bias toward own knowledge and beliefs; accuracy and honesty of responses; often must be used with other data collection techniques to improve quality of data collected.

27.3.1.2 Questionnaires

A questionnaire is a self-reporting query-based technique. Questionnaires are typically produced on printed paper, but due to recent technologies and in particular the Internet, many researchers engage in the use of online questionnaires, thus saving time, money, and eliminating the problem of a participant's geographical distance. There are three types of questions that can be used with questionnaires: (1) open questions, where the participants are free to respond however they like; (2) closed questions, which provide the participants with several choices for the answer; and (3) scales where the respondents must answer on a predetermined scale.

For online communities, questionnaire can be used to elicit facts about the participants, their behavior, and their beliefs/attitudes. Like interviews, questionnaires are an important technique for collecting user opinions and experiences they have had through the use of CMC and their overall existence in online communities.

The main advantages of questionnaires are as follows: they are faster to carry out than observational techniques; can cover low-probability events. Their disadvantages include the following: information is idealized version of what should rather than what does happen; responses may lack accuracy or honesty; danger of researcher bias toward subset of knowledge he/she possesses; must be used in conjunction with other techniques for validity.

27.3.1.3 Personas

Findings from interviews and questionnaires can be further used as a basis for developing user profiles using personas. A persona is a precise description of the user of a system and what he/she wishes to accomplish. (Cooper 1999). The specific purpose of a persona is to serve as a tool for software and product design. Although personas are not real people, they represent them throughout the design stage (Blomkvist 2002) and are best based on real data collected through query-based techniques.

Personas are rich in details, include name, social history, and goals, and are synthesized from findings through the use of query-based techniques with real people (Cooper 1999). The technique takes user characteristics into account and creates a concrete profile of the typical user (Cooper 1999).

For online communities, personas can be used to better understand the participants of the community and their background. Personas can also be used as a supplement to social network analysis (SNA; described in Section 27.3.4 in this chapter) to get a greater overview of the characteristics of key participants of a community. Using personas, web developers gain a more complete picture of their prospective and/or current users and are able to design the interfaces and functionality of their systems, to be more personalized and suited for the communication of the members of their online communities.

Advantages of personas include the following: can be used to create user scenarios; can be anonymous protecting use privacy; represent the user stereotypes and characteristics.

Disadvantages of personas include the following: if not enough personas are used, users are forced to fall into a certain persona type that might now accurately represent them; time-consuming.

27.3.2 Log Analysis

A log, also referred to as web-log, server log, or log-file, is in the form of a text file and is used to track the users' interactions with the computer system they are using. The types of interaction recorded include key presses, device movements, and other information about the user activities. The data are collected and analyzed using specialized software tools, and the range of data collected depends on the log settings. Logs are also time-stamped and can be used to calculate how long a user spends on a particular task or how long a user is lingering in a certain part of the website (Preece, Rogers, and Sharp 2002). In addition, an analysis of the server logs can help us find out when people visited the site, the areas they navigated, the length of their visit, the frequency of their visits, their navigation patterns, from where they are connected, and details about the computer they are using.

Log analysis is a useful and easy to use tool when analyzing online communities. For example, someone can use log analysis to answer more accurately questions such as student attendance of an online learning community. Furthermore, logs can identify the web pages users spend more time viewing and also the paths that they used. This helps identify the navigation problems of the website, but also gives a visualization of the users' activities in the virtual communities. For instance, in the case of e-learning communities, the log files will show which students are active in the CMC postings even if they are not active participants (few postings themselves), but just observing the conversations. Preece and Maloney-Krichmar (2003) notes that data logging does not interrupt the community, while at the same time can be used to examine mass interaction.

Advantages of logs (Preece, Rogers, and Sharp 2002) are as follows: helps evaluators analyze users' behavior; helps evaluators understand how users worked on specific tasks; it is unobtrusive; and large volumes of data can be logged automatically.

Disadvantages of logs (Preece, Rogers, and Sharp 2002) are as follows: powerful tools are needed to explore and analyze the data quantitatively and qualitatively; user privacy issues.

27.3.3 Content and Textual Analysis

Content analysis is an approach for understanding the processes that participants engage in as they exchange messages (McLoughlin 1996). There have been several frameworks created for studying the content of messages exchanged in online communities. Examples include work from Archer et al. (2001), Gunawardena, Lowe, and Anderson's (1997) model for examining the social construction of knowledge in computer conferencing, Henri's (1992) content analysis model, and Fahy, Crawford, and Ally's (2001) transcript analysis tool (TAT), which is described in more detail below.

The TAT focuses on the content and interaction patterns at the component level of the transcript (Fahy, Crawford, and Ally 2001). After a lengthy experience with other transcript tools and reviews of previous studies, Fahy, Crawford, and Ally (2001) chose to adapt Zhu's (1996) analytical model for the TAT. Zhu's (1996) model examines the forms of electronic interaction and discourse, the forms of participation, and the direction of participant interaction in computer conferences. The TAT also contains echoes of Vygotskian theory, primarily those dealing with collaborative sense-making, social negotiation, and proximal development (Cook and Ralston 2003). The TAT developers have come up with the following strategic decisions (Fahy 2003): the sentence is the unit of analysis; the TAT is the method of analysis; interaction is the criterion for judging conference success and topical progression (types and patterns).

The TAT was designed to permit transcript content to be coded reliably and efficiently (Fahy, Crawford, and Ally 2001). The advantages of TAT are (Fahy 2003; Cook and Ralston 2003; Fahy, Crawford, and Ally 2001) the following: it reveals interaction patterns that are useful in assessing different communication styles and online behavioral preferences among participants; it recognizes the complexity of e-conferences and measures the intensity of interaction; it enables the processes occurring within the conferences to be noted and recorded; it probes beyond superficial systems data, which mask the actual patterns of discussion; it relates usefully to other work in the area; it discriminates among the types of sentences within the transcript; it reflects the importance of both social- and task-related content and outcomes in transcript analysis research.

The unit of analysis of the TAT is the sentence. In the case of highly elaborated sentences, the units of analysis can be independent clauses, which, punctuated differently, could be sentences (Fahy 2003). Fahy, Crawford, and Ally (2001) have concluded that the selection of message-level units of analysis might partially explain problematic results that numerous researchers have had with previous transcript analysis work. They also believe that the finer granularity of sentence-level analysis results in several advantages (Fahy 2003): reliability; ability to detect and describe the nature of the widely varying social interaction, and differences in networking pattern, in the interactive behavior of an online community, including measures of social network density and intensity; confirmation of gender associations in epistolary/expository interaction patterns and in the use of linguistic qualifiers and intensifiers. Table 27.1 shows the TAT categories (Fahy, Crawford, and Ally 2001; Fahy 2003).

27.3.4 Social Network Analysis

"Social Network Analysis (SNA) is the mapping and measuring of relationships and flows between people, groups, organizations, computers or other information/knowledge processing entities. The nodes in the network are the people and groups while the links show relationships or flows between the nodes. SNA provides both a visual and a mathematical analysis of human relationships" (Krebs 2004, p. 1). Preece (2000) adds that it provides a philosophy and a set of techniques for understanding how people and groups relate to each other and has been used extensively by sociologists (Wellman 1982, 1992), communication researchers (Rice 1994; Rice et al. 1990), and others. Analysts use SNA to determine if a network is tightly bounded, diversified, or constricted; to find its density and clustering; and to study how the behavior of network members is affected by their positions and connections (Gartom, Haythornhwaite, and Wellman 1997; Scott 2000). Network researchers have developed a set of theoretical perspectives of network analysis. Some of these are (Bargotti 2000) as follows:

- Focus on relationships between actors than the attributes of actors
- Sense of interdependence: a molecular rather atomistic view
- Structure affects substantive outcomes
- Emergent effects

"The aim of social network analysis is to describe why people communicate individually or in groups" (Preece 2000, p. 183). The goals of SNA are (Dekker 2002) as follows:

- To visualize relationships/communication between people and/or groups using diagrams
- To study the factors that influence relationships and the correlations between them
- To draw out implications of the relational data, including bottlenecks
- To make recommendations to improve communication and workflow in an organization

Beidernikl and Paier (2003) list the following as the limitations of SNA:

- More theory that speaks directly to developers of online communities is needed.

TABLE 27.1
Transcript Analysis Tool Categories

Category

1: Questioning

The questioning category is further broken down into two types of questions:

1A: Vertical questions

These are questions that assume a "correct" answer exists and that they can be answered if the right authority to supply it can be found. Example: "Does anybody know what time the library opens on Saturdays?"

1B: Horizontal questions

For these questions, there may not be only one right answer. These questions invite negotiation. Example: "Do you really think mp3 files should become illegal or do you not see any harm by them?"

2: Statements

This category consists of two subcategories:

2A: Nonreferential statements

These statements contain little self-revelation and usually do not invite response or dialogue and their main intent is to impart facts or information. Example: "We found that keeping content up-to-date, distribution and PC compatibility issues were causing a huge draw on Ed. Centre time."

2B: Referential statements

Referential statements are direct answers to questions. They can include comments referring to specific preceding statements. Example: "That's right, it is the 1997 issue that you want."

3: Reflections

Reflections are significant personal revelations, where the speaker expresses personal or private thoughts, judgments, opinions, or information. Example: "My personal opinion is that it should not have been a penalty kick."

4: Scaffolding and engaging

Scaffolding and engaging initiate, continue, or acknowledge interpersonal interaction. They personalize the discussion and can agree with, thank, or otherwise recognize someone for their helpfulness and comments. Example, "Thanks Dave, I've been trying to figure that out for ages ☺."

5: References/authorities

Category 5 is compromised of two types:

5A: Quotations, references to, paraphrases of other sources.

Example, "You said, 'I'll be out of the city that day'."

5B: Citations, attributions of quotations, and paraphrases

Example: "Mathew, P. (2001). A beginner's guide to mountain climbing."

- The data collected may be personal or private.
- The analysis of the data is quantitative and specific to the particular network, whereas common survey data are qualitative and generalize answers on the parent population.

It is also worth pointing out that network analysis is concerned about dyadic attributes between pairs of actors (such as kinship, roles, and actions), whereas social science is concerned with monadic attributes of the actor (such as age, sex, and income).

There are two approaches to SNA as follows:

1. Ego-centered analysis: Focuses on the individual as opposed to the whole network, and only a random sample of network population is normally involved (Zaphiris, Zacharia, and Rajasekaran 2003). The data collected can be analyzed using standard computer packages for statistical analysis such as SAS and SPSS (Garton, Haythornthwaite, and Wellman 1997).
2. Whole network analysis: The whole population of the network is surveyed and this facilitates conceptualization of the complete network (Zaphiris, Zacharia, and Rajasekaran 2003). The data collected can be analyzed using microcomputer programs such as UCINET and Krackplot (Garton, Haythornthwaite, and Wellman 1997).

The following are important units of analysis and concepts of SNA (Garton, Haythornthwaite, and Wellman 1997; Wellman 1982; Hanneman 2001; Zaphiris, Zacharia, and Rajasekaran 2003; Wellman 1992):

- Nodes: The actors or subjects of study.
- Relations: The strands between actors. They are characterized by content, direction, and strength.
- Ties: Connect a pair of actors by one or more relations.
- Multiplexity: The more relations in a tie, the more multiplex the tie is.

- Composition: This is derived from the social attributes of both participants.
- Range: The size and heterogeneity of the social networks.
- Centrality: Measures who is central (powerful) or isolated in networks.
- Roles: Network roles are suggested by similarities in network members' behavior.
- Density: The number of actual ties in a network compared with the total amount of ties that the network can theoretically support.
- Reachability: To be reachable, connections that can be traced from the source to the required actor must exit.
- Distance: The number of actors that information has to pass through to connect one actor with another in the network.
- Cliques: Subsets of actors in a network, who are more closely tied to each other than to the other actor who is not part of the subset.

SNA is a very valuable technique when it comes to analyzing online communities, as it can provide a visual presentation of the community and, more importantly, it can provide us with qualitative and quantitative measures of the dynamics of the community. The application of SNA to the analysis of online communities is further demonstrated with a case study in Section 27.4 of this chapter.

27.4 CASE STUDIES

In this section, we present two case studies that demonstrate the use of theoretical and analytical techniques for studying online communities. In the first case study, we demonstrate how the results from an attitude toward thinking and learning questionnaire can be combined with SNA to describe the dynamics of a computer-aided language learning (CALL) online community. In the second case study, we present a theoretical activity theory model that can be used for describing interactions in online game communities.

27.4.1 Computer-Aided Language Learning Communities

In the first case study, we demonstrate a synthetic use of quantitative (SNA) and qualitative (questionnaires) methods for analyzing the interactions that take place in a CALL course. Data were collected directly from the discussion board of the "Learn Greek Online" (LGO) course (Kypros-Net Inc 2005).

LGO is a student-centered e-Learning course for learning Modern Greek and was built through the use of a participatory design and distributed constructionism methodology (Zaphiris and Zacharia 2001). In an ego-centered SNA approach, we have carried out an analysis of the discussion postings of the first 50 actors (in this case, the students of the course) of LGO.

To carry out the SNA, we used an SNA tool called "NetMiner" (Cyram 2004), which enabled us to obtain centrality measures for our actors. The "in and out degree centrality" was measured by counting the number of interaction partners per each individual in the form of discussion threads (e.g., if an individual posts a message to three other actors, then his/her out-degree centrality is 3, whereas if an individual receives posts from five other actors, then his/her in-degree is 5).

Because of the complexity of the interactions in the LGO discussion, we had to make several assumptions in our analysis as follows:

- Posts that received zero replies were excluded from the analysis. This was necessary to obtain meaningful visualizations of interaction.
- Open posts were assumed to be directed to everyone who replied.
- Replies were directed to all the existing actors of the specific discussion thread unless the reply or post was specifically directed to a particular actor.

In addition to the analysis of the discussion board interactions, we also collected subjective data through the form of a survey. More specifically, the students were asked to complete an Attitudes Towards Thinking and Learning Survey (ATTLS). The ATTLS measures, through the use of 20-Likert scale questions, the extent to which a person is a "connected knower" (CK) or a "separate knower" (SK). People with higher CK scores tend to find learning more enjoyable and are often more cooperative, congenial, and more willing to build on the ideas of others, whereas those with higher SK scores tend to take a more critical and argumentative stance to learning (Galotti et al. 1999).

The out-degree results of the SNA are depicted in Figure 27.1 in the form of a sociogram, and the in-degree results are depicted in Figure 27.2. Each node represents one student (to protect the privacy and anonymity of our students, their names have been replaced by a student number). The position of a node in the sociogram is representative of the centrality of that actor (the more central the actor the more active). As can be seen from Figure 27.1, students S12, S7, S4, S30 (with out-degree scores ranging from 0.571 to 0.265) are at the centre of the sociogram and possess the highest out-degree. The same students also posses the highest in-degree scores (Figure 27.2). This is an indication that these students are the most active members of this online learning community, posting and receiving the largest number of postings. In contrast, participants in the outer circle (e.g., S8, S9, S14, etc.) are the least active with the smallest out-degree and in-degree scores (all with 0.02 out-degree scores).

In addition, a clique analysis was carried out (Figure 27.3), and it showed that 15 different cliques (the majority of which are overlapping) of at least three actors each have been formed in this community.

As part of the ego-centered analysis for this case study, we look in more detail at the results for two of our actors: S12, who is the most central actor in our SNA analysis, that is, with the highest out-degree score, and S9, an actor with the smallest out-degree score. It is worth noting that both members joined the discussion board at around the same time.

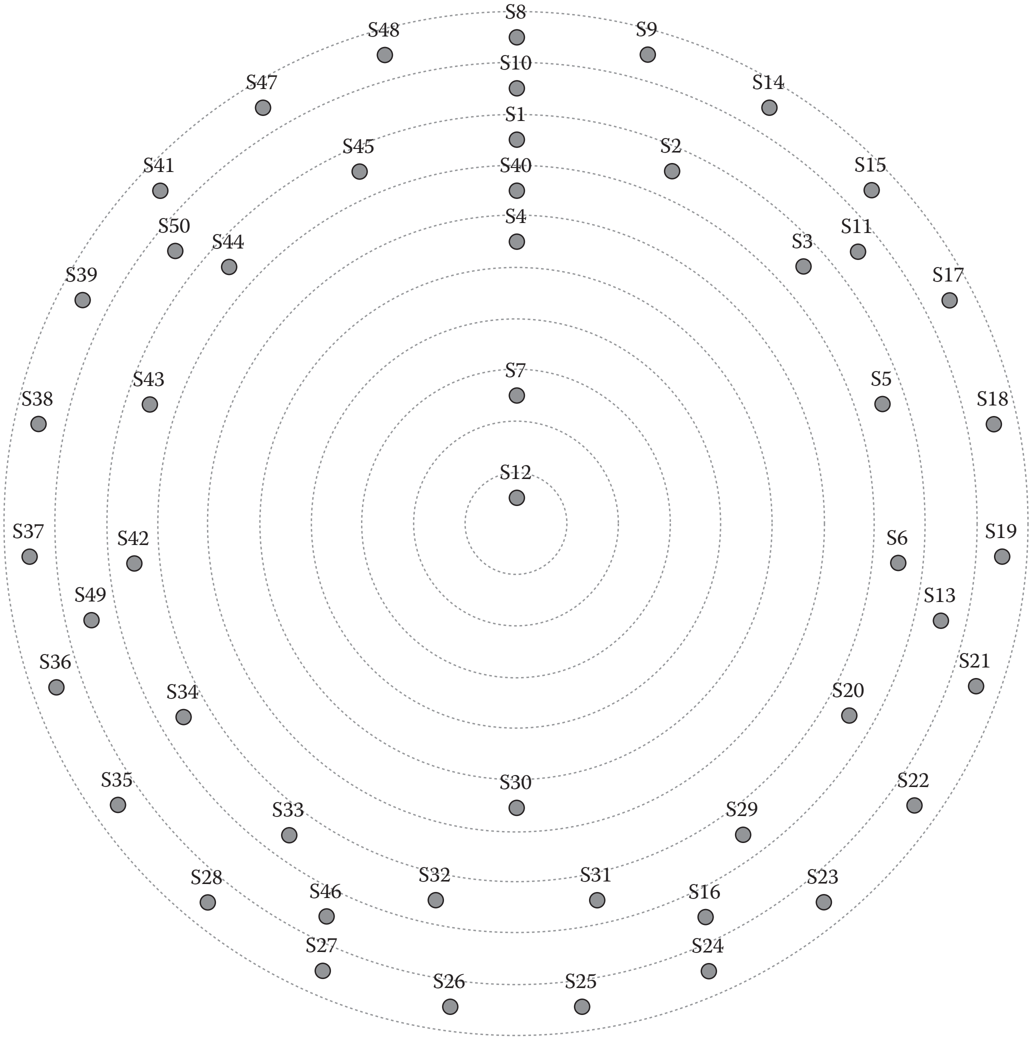


FIGURE 27.1 Out-degree analysis sociogram.

First, through a close look at the clique data (Table 27.2), we can see that S12 is a member of 10 out of the 15 cliques, whereas S9 is not a member of any, an indication of the high interactivity of S12 versus the low interactivity of S9. In an attempt to correlate the actors' position in the SNA sociogram with their self-reported attitudes toward teaching and learning, we looked more closely at the answers these two actors (S12 and S9) provided to the ATTLS. Actor S12 answered all 20 questions of the ATTLS with a score of at least 3 (on a 1–5-Likert scale), whereas S9 had answers ranging from 1 to 5. The overall ATTLS score of S12 is 86, whereas that of S9 is 60. A clear dichotomy of opinions occurred on 5 of the 20 questions of the ATTLS. S12 answered all 5 of those questions with a score of 5 (strongly agree), whereas S9 answered them with a score of 1 (strongly disagree). More specifically, S12 strongly agreed on the following:

1. She or he is more likely to try to understand someone else's opinion than to try to evaluate it.
2. She or he often finds herself or himself arguing with the authors of books read, trying to logically figure out why they're wrong.
3. She or he finds that he or she can strengthen his or her own position through arguing with someone who disagrees with them.
4. She or he feels that the best way to achieve his or her own identity is to interact with various other people.
5. She or he likes playing devil's advocate—arguing the opposite of what someone is saying.

S9 strongly disagreed with all the above statements. These are all indications that S12 is a CK, whereas S9 is a SK.

This case study showed that the combination of quantitative and qualitative techniques can facilitate a better and deeper understanding of online communities. SNA was found to be a very useful technique for visualizing interactions and quantifying strengths and dynamics in online communities. In combination with the ATTLS, it was possible to

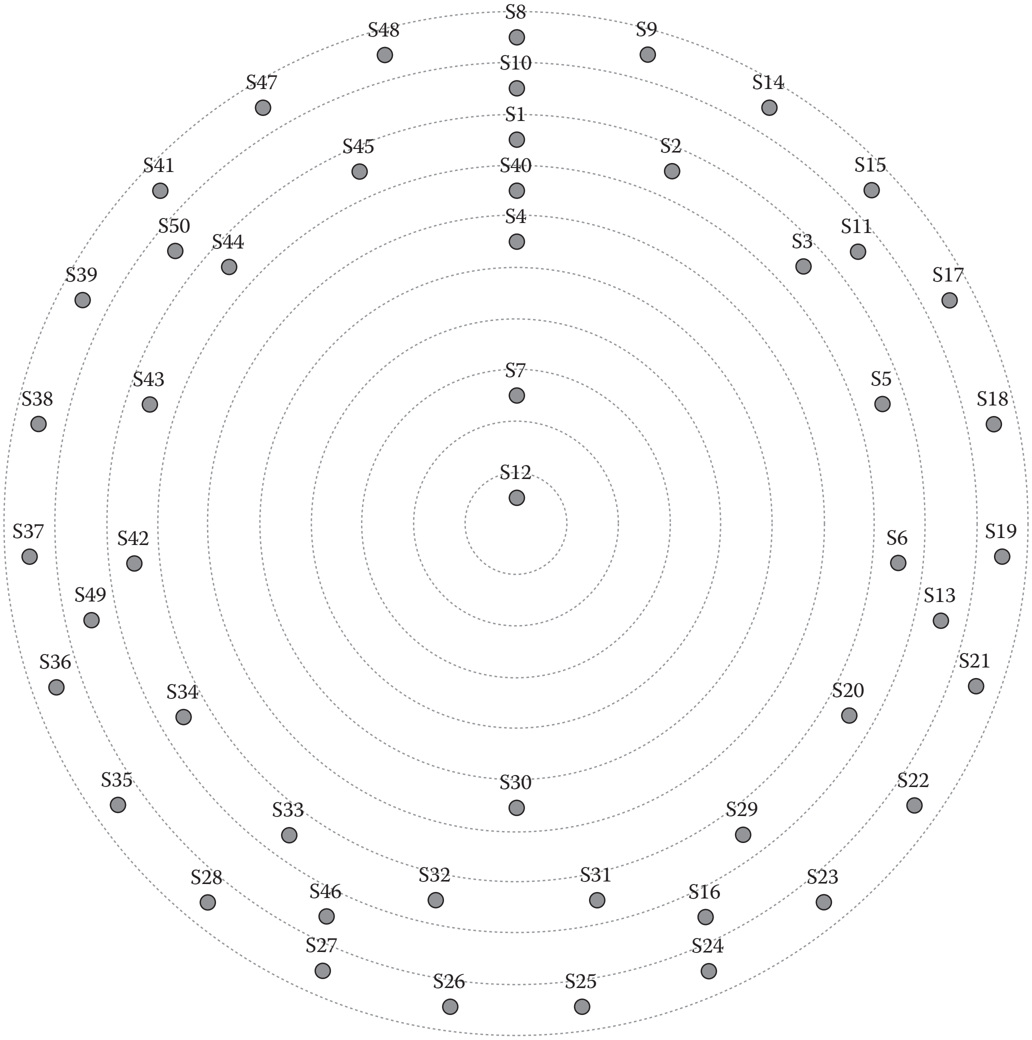


FIGURE 27.2 In-degree analysis sociogram.

identify the key players of the e-learning community. These members' roles show them to be more powerful and central in the discussions. Identifying their characteristics enables us to make re-enforcements to the community by making other participants more active in the discussion board communication. This active learning approach could in-turn improve the pedagogical value of e-learning within these communities.

27.4.2 Game Communities and Activity Theoretical Analysis

The main motivation of the second case study arises from the more general area of computer game-based learning. Game-based learning has focused mainly on how the game itself can be used to facilitate learning activities, but we claim that the educational opportunity in computer games stretches beyond the learning activities in the game per se. Indeed, if you observe most people playing games, you will likely see them download guidelines from the Internet and participating in online forums to talk about the game and share strategies. In actuality, almost all game playing could be described as a social experience, and it is rare for a player to play a game alone in any meaningful sense (Kuo 2004). This observation is even more evident in MMORPGs, which has been discussed in Section 27.2.3.1. For example, the participation in a MMORPG is constituted through language practice within the in-game community (e.g., in-game chatting and joint task) and out-of-game community (e.g., the creation of written game-related narratives and fan-sites). The learning is thus not embedded in the game, but it is in the community practice of those who inhabit it.

27.4.2.1 Types of Game Communities

Therefore, we believe that the study on computer games should be expanded to include the entire game community. Computer game communities can be categorized into three classes, which we have identified (Figure 27.4; Ang, Zaphiris, and Wilson 2005) as follows:

Single Game–play Community: this refers to a game community formed around a single player game. Although players of a single player game such as The Sims 2 and Final Fantasy VII play the game individually, they are associated with an out-of-game community, which discusses the game either virtually or physically.

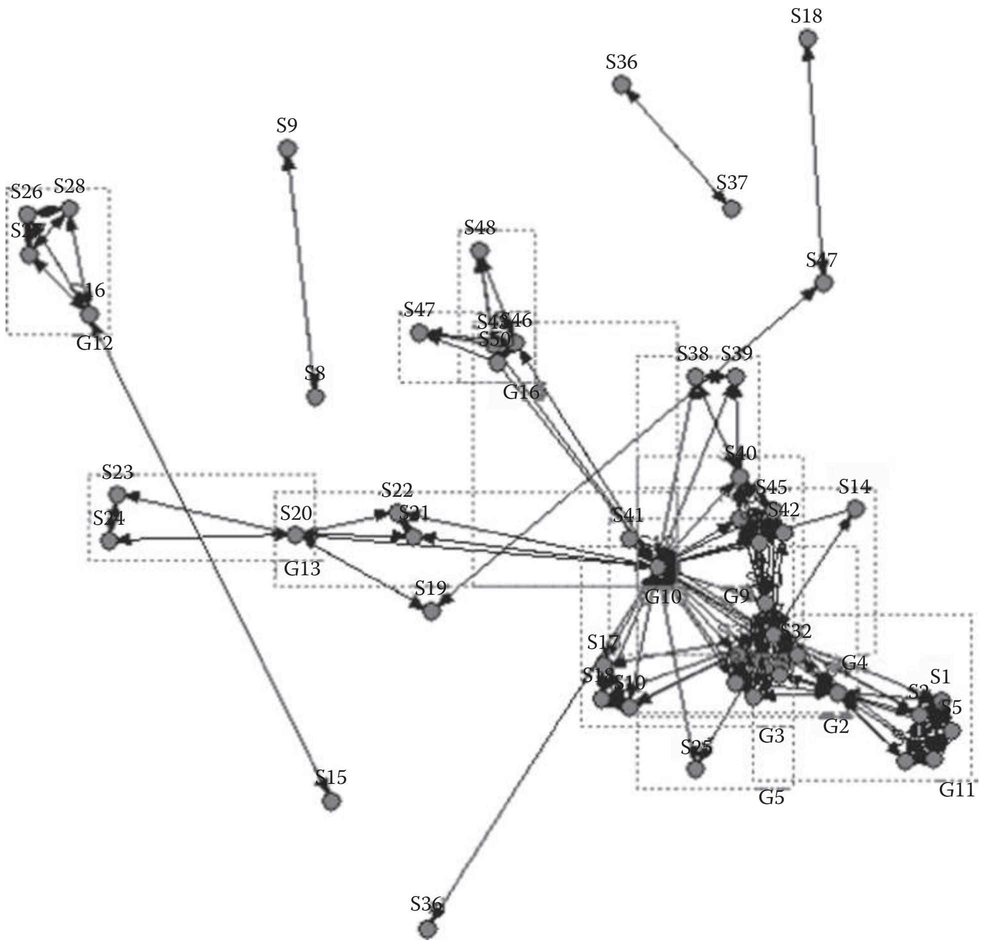
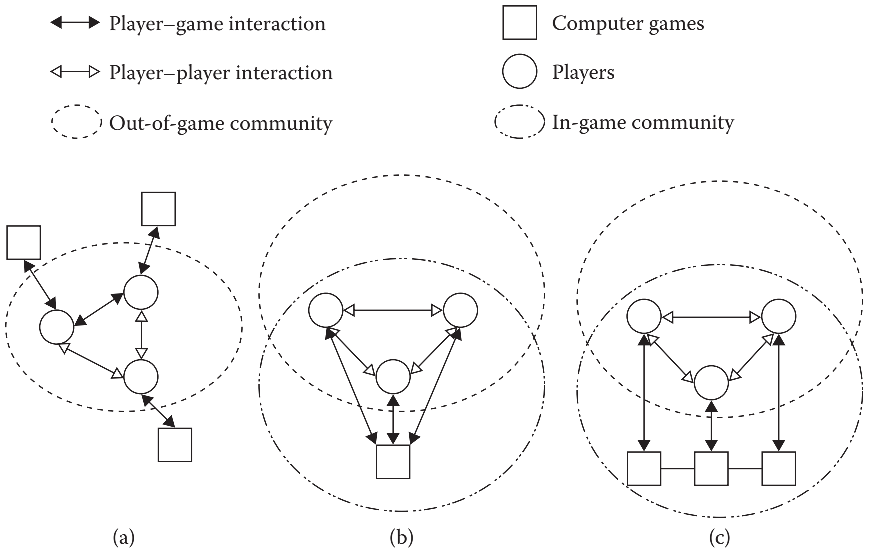


FIGURE 27.3 Clique analysis sociogram.

TABLE 27.2
Clique Analysis of the Learn Greek Online Discussions

Cliques	Actors
K1	S12, S7, S30, S40, S43, S44, S45
K2	S12, S7, S30, S4
K3	S12, S7, S10, S11, S13
K4	S12, S7, S14
K5	S12, S7, S25
K6	S12, S7, S41
K7	S12, S20, wS21, S22
K8	S12, S29, S4, S30, S31, S32, S33, S34
K9	S12, S38, S39, S40
K10	S12, S46, S49, S50
K11	S1, S2, S3, S4, S5, S6, S7
K12	S16, S26, S27, S28
K13	S23, S20, S24
K14	S47, S46, S49, S50
K15	S48, S46, S49, S50

Social Game–play Community: this refers to multiplayer games, which are played together in the same physical location. It creates game communities at two levels: in-game and out-of-game. Occasionally, these two levels might overlap. The out-of-game interaction might be affected by issues beyond the specific game system; for example, the community starts exchanging information about another game.



(a) Single game–play community (b) Social game–play community (c) Distributed game–play community

FIGURE 27.4 Types of game communities.

Distributed Game–play Community: this is an extension of social game–play community, but it emphasizes the online multiplayer game in which multiple sessions of game are established in different geographical locations.

The study of game communities, especially out-of-game communities, from the perspective of education is still very much unexplored. We believe the potential of games in education is not limited to what is going on in the game. Educators could benefit by studying games as a social community because games are now becoming a culture that permeates the life of everyone, especially the younger generation. Black (2004) has investigated the interactions among participants in a virtual community of Japanese comic fans, which involve a lot of reading and writing throughout the site. She examines how the community of fans helps each other with English language writing skills and with cross-cultural understanding. In this section, we have pointed out that game communities can emerge from both single-player and multiplayer games. We believe that by further studying the social interaction in the game community, we will be able to utilize games in learning in a more fruitful way. In Section 27.4.2.2, we apply and evaluate one of these models of game communities to a specific scenario in knowledge building using activity theory.

27.4.2.2 Activity Theory

In this case study, we demonstrate how activity theory can be used to analyze an out-of-game community around a single-player game, which is based on constructionist activities as proposed by Papert (1980). Papert claims that even for adults, learning remains essentially bound to context, in which knowledge is shaped by the use of external supports. His approach helps us understand how learning is actualized when individual learners construct their own favorite artifacts or object-to-think-with (Papert 1980).

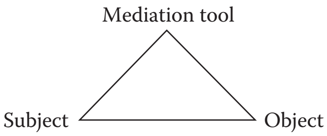


FIGURE 27.5 Vygotsky's mediation.

Although Papert's theory provides a solid framework for understanding children's and even adults' ways of learning by designing, it does not offer a systematic framework for analyzing the construction activities within a learning community. Analyzing constructionist activities can be useful as it could help inform constructional design for educational purposes. The most significant analysis includes the learning within a community and the development of an individual. We are also interested in finding out how tools such as computers help learners construct artifacts and knowledge. Hence, we would like to draw from the Vygotskian naturalist approach, which emphasizes human activity systems. Vygotsky (1930) formulated a theoretical concept that is very different from the prevailing understanding of psychology, which was dominated by behaviorism at that time. This new orientation was a model of tool-mediation and object-orientedness. He proposes the classic triangle model to demonstrate the idea of mediation:

In Figure 27.5, the subject is the individual engaged in the mediated action, the mediating artifact or tool could include physical artifacts and/or prior knowledge of the subject. The object is the goal/objective of the activity. Although constructionist learning relies very much on computational tools, the concept of mediation is not explicated. Figure 27.5 shows explicitly that the relationship between the subject and the object is no longer straightforward; instead, it is mediated by the tool. For example, when building a website, the subject is working toward an objective (e.g., to add a table in the webpage) using not only the computer (external tools) but also her internal understanding of how websites and computers work (internal tools).

TABLE 27.3
Hierarchy of Activity

Unit of Analysis	Stimulus	Subject	Language Learning Example
Activity	Object	Community	Engage in a meaningful conversation
Action	Goal	Individual	Sentence construction
Operation	Conditions	Unconscious	Word selections, grammar rules

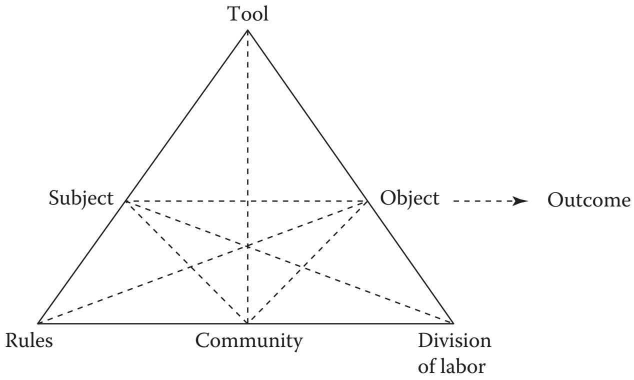


FIGURE 27.6 The triangle activity system diagram.

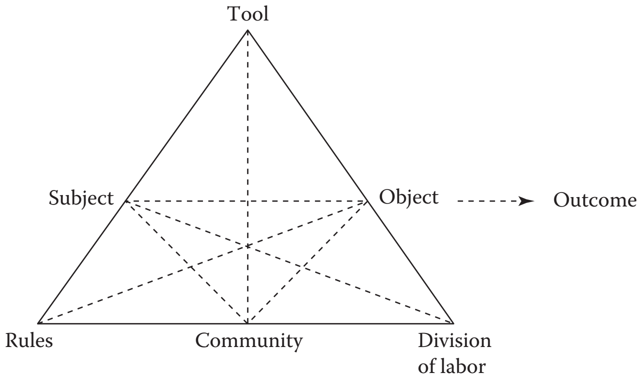


FIGURE 27.7 The transformation of individual action-operation.

Leont'ev (1978) extends this notion of activity to differentiate between an individual action and a collective activity by proposing a hierarchy of activity (Table 27.3). Collective activity is connected to the object of the whole community, of which each individual subject is often not consciously aware. An individual action is connected to a conscious goal. Below the collective activity and individual action, there is a level of operations dependent on the conditions in which the action is performed. Thus, an activity system can be analyzed at three levels: (1) the activity level, which is oriented toward the object/objective and carried out by the whole community; (2) the action level, which is directed at the individual goal; and (3) the operation level, which is elicited by conditions and is performed unconsciously.

This hierarchy is crucial in explaining the learning process in an activity system. We would like to illustrate an example of this hierarchy in learning a foreign language (Table 27.3). The overall objective is to be able to engage in a meaningful conversation. In the beginning, the learner has to work on the grammar and the choice of words at a conscious level. When the learner has reached a higher proficiency level, these actions are transformed into operations. The learner no longer needs to select appropriate words and check grammar rules deliberately as these have been learned thoroughly and are now operating unconsciously. The consciousness of the learner is now focused on expressing himself properly depending on the objective of the conversation. Grammatical rules become invisible to the learner, and he is only selecting appropriate goals to be achieved. Therefore, it can be inferred that activity theory treats learning as the shift from the higher level (action) to the lower level (operation) in the hierarchy.

Drawing on work by Vygotsky and Leont'ev, Engeström (Engestrom 2001) views all human activities as contextualized within an interdependent activity system. Engeström adds collective mediation to Vygotsky's tool mediation and presents the triangle model of activity system (Figure 27.6).

In the diagram, the subject is the individual or a group who are selected as the point of view of the analysis. The object refers to the raw material or the problem space at which the activity is directed and which is transformed into outcomes with the help of external and internal tools. Tools are the concepts, physical tools, artifacts, or resources that mediate a subject's interactions with an object. The community refers to those with whom the subject shares the same general object. The division of labor (DOL) is the classification of tasks among the members on the community, whereas the rules are the regulations, norms, and conventions within the activity system.

Constructionist learning can be described and visualized through this activity triangle. Mediated by the tool and the community, the learner externalizes her initial stage of knowledge through object construction. The individual externalization (mediated by the tool) can be broken down into actions and operations. Actions are directed toward a personal goal and are carried out with careful deliberation. For example (Figure 27.7), in a book-writing activity, the author (an expert word processor user) will operate (e.g., typing) the word processor at the unconscious level and consciously acts on the book (to select appropriate words, construct meaningful sentences, and paragraphs) she is writing. At a certain point, the author encounters a new condition with the word processor, which she is not familiar with: say to insert a table into the book. Under this new condition, a breakdown is said to have happened. The conscious effort of the author is not placed anymore on the book itself but instead is now placed on the word processor (e.g., to achieve the action [insert tables], the author performs the operation [read help files]). Once the author has thoroughly learned about the table insert, she can again act on the book consciously and development is said to have happened.

27.4.2.3 Activity Theoretical Analysis for Constructionist Learning

We have conducted a study to analyze a wiki-based game community using activity theory. The goal of this study was to demonstrate the usefulness of activity theory in researching online communities. The collaborative construction activity in a wiki-supported community devoted to writing a game guidebook is examined. Activity theory is used as an analytical tool to investigate this community. In this section, experiences and challenges from the analysis are reported to give some insights into how activity theory can be helpful for online community research.

The analysis on online communities can be done through the lens of various concepts associated with activity theory such as the levels of activity hierarchy and different perspectives of the triangle model as explained in Section 27.4.2.2.

For instance, we can start our analysis with the most basic aspect of the constructionism by simply examining the relationship between subjects and the object. Then, we can analyze Vygotsky's mediation model of activity system consisting of individual actions and tools. The analysis is also possible to be extended to the whole community of this system to include emerging rules and DOL that mediate the community. The focus can also be placed mainly on the constructionist concept of externalizing the internal meanings onto a sharable artifact through mediation. More specifically, we can look into (but are not limited to) these aspects:

- Subject–object: What are the constructionist actions that act on the object and transform the object into outcome?
- Subject–tool–object: How do actions shift to operations and vice versa? How do tools mediate individual actions and operations? What is the nature of the mediating tools? How do they support knowledge building?
- Subject–rules–object and subject–DOL–object: What is the nature of implicit and explicit rules? How is DOL manifest in the community? How do rules and DOL support knowledge building?

Apart from its focus on both individual and collective development through action–operation transformation, activity theory also helps analyze the tools, capture the rules and the DOL, which mediates these actions. These must be further explained to differentiate individual mediation and collective mediation. Taking the example of our study on game community, individual mediation places its emphasis on "how a user uses the tool to write the game guide, without taking into account how other users act in the community." In other words, it is about the affordance of the tool to support what an individual can do.

Collective mediation is about the community, which consists of two major components: rules and DOL. Rules define what can be done and cannot be done in a community. This should not be confused with the affordance of the tool. The tool might afford certain actions such as writing in abusive language, but the rules might want to ban this action. DOL is self-explained: how the work load is divided among many users in a community.

Activity theory appears to be a promising framework as it gives an analytical lens on analyzing and interpreting the data. Activity theory provides different perspectives of analysis, as it casts different light on the collected data as researchers can examine it from many perspectives by focusing on different subtriangles of the activity system diagram. It also helps us examine learning process: how learning occurs individually and collectively through the transformation of hierarchy of activity from action to operation. Furthermore, both individual and collective aspects are given equal importance. Activity theory informs the development of the whole community and the individual development. It explains how individual development contributes to the community growth and vice versa.

On the basis of our study, individual actions help sharpen the mediation tool, whereas collective actions bring about new rules or refine existing rules that mediate the collective action. In short, activity theory is useful to analyze the community in the following ways:

- It helps understand the individual mediation process: subject–tool–object
- It clearly presents the collective mediation process: subject–community–object
- It reveals the emerging rules and DOL in the community

In a wiki space, knowledge is socially constructed; it is created individually with tools, negotiated, and agreed within a community based on emerging rules and DOL. It starts as a single unit of information (a page in this context) and grows organically and evolves into a complex and well-structured set of knowledge. From our findings, we induce what contributes to the development of the community:

- Users share some historical backgrounds: they already share some of the tools/rules before joining the community, and they also share the interest on the same game.
- Users share the same object (goal), which is to build a game guide book.
- A user's individual action: this goal-oriented individual action triggers negotiations that lead to the growth of the space.
- The community's agreement on the object: not only share the same object, the community must be able to negotiate and agree on the object.
- Tools that support these actions and negotiations.
- Emerging rules that coordinate the activity.
- DOL that divide the responsibilities.

The evolution of a knowledge-building community needs more than a group of devoted users who share the same object. It involves negotiation and agreement among the users on the object. Although every user tends to act toward their own goal, it takes the compromise of the entire community to agree on the object.

Activity theory helps online community researchers identify design issues at two aspects: the software application as tools and the social interaction within the community around the tool. It also reveals the development of the community-building process from the individual and collective level through the shift of activity hierarchy. We thus believe that analyzing online communities with activity theory will yield fruitful results and give insights on online community design.

Finally, we must reiterate the fact that activity theory itself is not limited to what is presented in the triangle diagram as proposed by Engeström. Although his model is tremendously useful, it overlooks several significant concepts of activity theory. One major limitation is the static representation of activity theory. The triangle diagram represents only a snapshot of a particular time, thus making it hard to analyze the activity across time. It is understood that Engeström's model is intended to be open so that it can be used in various domains, but this has proven to pose a serious difficulty among the practitioners as some researchers have started to operationalize it so that it is more practical in day-to-day methodology (Korpela, Soriyan, and Olufokunbi 2000; Barab, Hay, and Yamagata-Lynch 2001; Mwanza 2002). Hopefully, with the increasing attention drawn by activity theory, the theory will be expanded and operationalized to fit the purpose of HCI research in general and online community in particular.

27.5 ANALYZING AND DESIGNING ONLINE COMMUNITIES WITH SOCIAL SIMULATION

Developing online communities involves usability design, which supports consistent, controllable, and predictable human–computer interaction, and sociability design, which focuses on human–human interaction (i.e., social interaction) and social policies (Preece 2000).

The majority of research on sociability design has so far been revolved around conventional HCI methods such as query-based techniques (e.g., questionnaires, interviews), observation, and content analysis to identify the sociability requirements of the online community through various aspects: purpose, people, and policies (Preece 2000).

For instance, several studies have already analyzed the content of messages that people post in online communities (Savicki, Lingenfelter, and Kelley 1996; Klemm et al. 1999; Golder and Donath 2004) and constructed typology of information exchange that described people's online behavior (Burnett 2000). Others investigated and contrasted the interaction patterns in different types of communities (Turner et al. 2005). Using query-based techniques, some have examined people's motivation of certain online behavior such as altruistic helping behavior in an open-source software development community. (Raymond 1999; Kollock 1999; Niedner, Hertel, and Hermann 2000).

These methods often aim to generate a rich description of the current state of the community, which can then be transformed into needs and requirements so that a social software system can be developed to facilitate users' needs and activities.

When developing online communities, we believe that it is useful to be able to compare design alternatives. For instance, how different social policies will lead to the growth of the community. In addition, we would also like to find out how changes in an existing community will affect its future trend.

However, online communities are complex phenomena as they encompass humans interacting with each other, exchanging and diffusing information. Such a system is dynamic and complex, continually changing and evolving.

Current research practice on (online) community offers little opportunity for causal analysis (Jager, Popping, and Sande 2001). Interaction patterns leading to the emergence of different types of social networks are difficult to study experimentally.

One possible way of making comparisons is through simulation. Suppose we have an online community of elderly people on the topic of depression and we notice that there are too many off-topic discussions. We want to introduce a new policy/new design that would reduce off-topic discussion, but we are not sure whether doing so will be beneficial or detrimental to the future growth of the community. Often it is impractical to run a large-scale social experiment on such a massive online community. Furthermore, it could involve complicated ethics issues when conducting experiments that deal with this sensitive population. In this case, we could develop a social simulation that mimics the interaction behavior of the existing community and conduct "virtual" experiment with the simulation.

In other words, simulations can provide a test-bed for what-if situations, which cannot be studied otherwise through other methods.

27.5.1 Agent-Based Model

In terms of social simulation (Gilbert and Troitzsch 2005), an agent-based model (ABM) is a computational model for studying the process of the social system as a whole through simulating the actions and interactions of autonomous individuals, known as agents. Agents can include a whole range of social actors with data-gathering and decision-making ability, sometimes equipped with sophisticated learning capabilities and are adaptive to the environment.

The ABM described in this chapter simulates the interaction behaviors of individual members of an online community to understand the dynamics of the social network in the online community and to understand how various interventions affect community performance.

ABMs are models between natural language and mathematical symbols. They model reality with formal language (i.e., programming language), which is understood by the computer.

Agents impose four key assumptions (Wooldridge and Jennings 1995).

- They are autonomous: the system is not modeled as a globally integrated entity. The global patterns emerge from local interaction among autonomous decision-making agents.
- They are interdependent: agents are influencing each other's behavior directly or indirectly.
- They follow simple rules: each individual is modeled with simple behavior, which could produce complex global patterns.
- They are adaptive: agents could learn from their past experience, thus alter their behavior.

The global process of online communities is distributed since there is limited/no central control and the result depends largely on the interactions between agents. Hence, agent-based modeling approach is bottom-up: it is a nonlinear system, which is more than the sum of their parts (Waldrop 1992). The macro-behavior can be different from the underlying micro-motives.

The goal of such modeling is often to reproduce macro-events from the bottom-up and to obtain emergent regularities.

We maintain that ABM can benefit sociability design of online communities. Most importantly, the models can help define good design, that is, how a given design decision is positive for the community. By generating many runs with different conditions, large data sets of online communities can be accumulated.

Traditionally, sociologists have understood social life as a system of institutions and norms that shape individual behavior from top-down. From the view point of SNA, Haythornthwaite and Wellman (1998) argued that individuals' behavior is affected by the kind of ties and the network in which they are involved than by attributes they possess. We believe that the reverse is also true, in which the ties and the network are affected by the individuals' behavior. In this study, we developed an ABM to study the effect of player interaction on the formation of social networks of the guild community from bottom-up.

Dealing with linear systems, the behavior of the whole system corresponds exactly with the sum of its constituting parts (such as multiple regression models). In nonlinear systems, even if the observer has a good understanding of how each part works, it will not be possible to understand the system as a whole. Instead of starting from the system as a whole and decompose it (top-down), it will be more fruitful to start from its constitutive parts (bottom-up). In such systems, coherent behaviors not defined a priori may spontaneously emerge from the aggregate dynamics. Even a simple ABM can exhibit complex behavior patterns and provide valuable information about the dynamics of the real-world system that it emulates.

ABM simulation can work like a virtual laboratory through which the researcher can test hypotheses, manipulating variables and constraints of the model and observe the outcome. Therefore, ABMs are a powerful tool in theoretical development and explanation. We can use ABMs to explore plausible mechanisms underlying the observed patterns.

Research using ABM has demonstrated the ability to grow societies (Epstein and Axtell 1997). Various social phenomena have been studied using ABM, including the simulation of digital markets (López-Sánchez et al. 2004), violence and conflict modeling (Epstein 2002; Jager, Popping, and Sande 2001), trust and cooperation (Macy and Sato 2002), archaeological simulation (Doran et al. 1994), ecosystem management (Antona et al. 2002), and so on. Clearly, not all social phenomena can be meaningfully modeled through the bottom-up approach. Therefore, ABMs are most appropriate for examining processes that lack central coordination.

27.5.2 Simulating Social Networks

The main strength of ABM lays in its focus on relational simulation for theoretical research that bridge between micro and macro levels of analysis. Therefore, not surprisingly, ABM has been applied to simulate interaction of agents in social networks. In most cases, AMB is applied to study two situations in social networks. First, simulations are developed to examine agents' interaction and the aggregate interaction dynamics within specific topology of social networks. Wilhite (2001), for instance, examined trading interaction, that is, search, negotiation, and exchange within four types of social network topology: global network, local disconnected network, local connected network, and small-world network. Instead of exploring the process of network formation, he was interested in understanding how different types of social network configurations affect trading behaviors.

More common in the research of social network simulation is the generation of social network based on agents' interactions. For example, some work has been carried out by Ahrweiler, Pyka, and Gilbert (in press), Lamieri and Ietri (2004), and Özman (2007) to explore the network formation based on the simulation of innovation creation and knowledge diffusion in the industry. In another study, Zhang, Ackerman, and Adamic (2007a,b) used simulations to replicate the structural characteristics of an online Java discussion forum and to evaluate how different types of expert ranking algorithm may perform in communities with different characteristics. Apart from these, ABM has also been used in studying network evolution over time (Carley, Ju-Sung, and Krackhardt 2001; MaxTsvetovat and Carley 2002).

27.5.3 Application to Online Community Design

One obvious question is of course "how do we practically relate design/policy decisions to the simulation parameters." We believe this is where conventional HCI methods meet social simulations.

Existing HCI methods that focus on empirical study can be easily integrated into simulation study. Experimental studies can be carried out to examine the effect of a policy change/graphical user interface element on interaction behavior at individual/micro level. For instance, we could conduct a small-scale experiment with 30 people and estimate how much a particular "reward system" (a design decision) increase/decrease their initiation of acting (a simulation parameter). Then, we can plug in this empirical result into the simulation to predict the social network that might arise through the introduction of this reward system.

27.5.4 Simulations Are a Powerful Technique for Understanding Sociability in Online Communities

Since we are unable to directly modify the community, we need an alternative way to study how the underlying characteristic of the community influences the social network that arises. We believe that simulations allow us to understand this causal relationship by generating data that may not be obtainable empirically.

This understanding can also enable us to design better ways to transform and expand these communities with minimal risks.

27.6 DISCUSSION AND CONCLUSIONS

This chapter looked at the definitions of CMC and online communities and pointed out the multidisciplinary nature of the definitions and the way online communities are being analyzed and studied. In Section 27.2, we introduced the different types of CMC and online communities. In Section 27.3, we provided an overview of some of the most commonly used techniques and theoretical frameworks for analyzing online communities. Then in Section 27.4, we used two case studies to demonstrate the use of those techniques. Online communities are widely used for general entertainment and current affairs discussions as well. Professional communities (e.g., business, art, and industry specific) are also being formed within the online environment.

The study of online communities is flourishing, primarily due to the increasing popularity of online services and tools that provide the construction of such networks of users. The study of such complex communities requires the use of a synthesis of methodologies and theoretical foundations. In our first case study, we demonstrated how SNA can be used to model and visualize online community interactions; in the second one, the theoretical foundations based on activity theory were applied to the domain of game-based communities.

At the beginning of the Internet technology, online communities were solely used for synchronous (often just chat) or asynchronous (most commonly discussion board) textual interaction. This is no longer the case as people are interacting in online communities using new and more innovative interaction paradigms. Game-based communities, for instance, allow users to represent themselves (through the use of 3D avatars) visually in virtual environments, which are often depicted as a fantasy and unrealistic world. They can navigate, change, or even create the virtual world (and thus the context of the community) they interact with. As such, the traditional boundary between author and reader is distorted as the designer (authors) is not the only one who determines what the system is like. Rather, the participants (readers) are co-constructing the virtual world as they are not anymore using the tool to communicate; they are creating and interacting with a virtual environment through which they can meet, socialize, and work with others. Preece (2000) was the first to identify and stress this important social dimension of online communities. Now, this online sociability is becoming more mature and more central to the online communities with which we interact.

This new paradigm of interaction poses new challenges for researchers and practitioners. The importance of appropriate representation of emotions and other social cultural cues in online communities is now becoming even more important. With textual interfaces, there was an attempt to represent these social cues through the use of emoticons. How can this, for example, be transferred to the domain of virtual game communities? Do we want our avatars to behave like us or do we want them to have some alternate and illusionary identities with extraordinary abilities or unusual behaviors? How can we come up with new interfaces and new interaction paradigms that can facilitate this better, in order to cope with the new demands from the users in such online communities?

A second area that is gaining popularity is the research of online communities, or Internet research. Content analysis and query-based techniques were sufficient if what we wanted was a first good impression of the interactions taking place in an online textual community. This is not anymore the case. Studying the sociability of a game-based community, for example, requires the synthesis of more techniques. We might want to immerse ourselves in that community, engage in long-term ethnographic studies of its environment, and get the first-hand experiences of what happens in it. We might have to use a social cultural theoretical framework (e.g., activity theory) to get a better understanding of the way people behave, differently from the real world, in these inherently social environments. We might have to quantify, through the use of SNA for example, the dynamics of the networks and subgroups that evolve around these communities.

The possibility of online communities is unlimited with the emergence of more mature and imaginative virtual worlds. Only by treating them in equality with their physical counterparts, which encapsulate the practices of economy, politics, ideology, and everyday life, can we research and study them intellectually.

REFERENCES

Ahrweiler, P., A. Pyka, and N. Gilbert. 2004. Simulating knowledge dynamics in innovation networks (skin). In *Industry and Labor Dynamics: The Agent-Based Computational Economics Approach*, ed. R. Leombruni and M. Richiardi. Singapore: World Scientific Press.

Ang, C. S., P. Zaphiris, and S. Wilson. 2005. *Social Interaction in Game Communities and Second Language Learning*. Edinburgh, U.K.: The 19th British HCI Group Annual Conference.

Antona, M., P. Bommel, F. Bousquet, and C. L. Page. 2002. Interactions and organization in ecosystem management: The use of multi-agent systems to simulate incentive environmental policies. Paper presented at the 3rd Workshop on Agent-Based Simulation. Ghent, Belgium.

Archer, W., R. D. Garrison, T. Anderson, and L. Rourke. 2001. A framework for analyzing critical thinking in computer conferences. European Conference on Computer-Supported Collaborative Learning, Maastricht, The Netherlands.

Aronsson, L. 2002. Operation of a Large Scale, General Purpose Wiki Website. Experience from susning.nu's first nine months in service. In *Proceedings of the 6th International ICCC/IFIP Conference on Electronic Publishing*, ed. J. Á. Carvalho, A. Hübler, and A. A. Baptista, 27–37. Karlovy Vary, Czech Republic. Berlin: Verlag für Wissenschaft und Forschung.

Barab, S. A., K. E. Hay, and L. C. Yamagata-Lynch. 2001. Constructing networks of action-relevant episodes: An in situ research methodology. *J Learn Sci* 10(1 & 2):63–112.

Beidernikl, G., and D. Paier. 2003. Network analysis as a tool for assessing employment policy. In *Proceedings of the Evidence-Based Policies and Indicator Systems Conference 03*. London.

Black, R. W. 2004. Anime-inspired affiliation: An ethnographic inquiry into the literacy and social practices of English language learners writing in the fan-fiction community. Paper presented at 2004 meeting of American Educational Research Association, San Diego, CA.

Blomkvist, S. 2002. Persona—an overview, Uppsala University http://www.it.uu.se/edu/course/homepage/hcinet/ht04/library/docs/Persona-overview.pdf (accessed November 22, 2004).

Borgatti, S. 2000. *What is Social Network Analysis*. Retrieved on November 9, 2010 from http://www.analytictech.com/networks/whatis.htm.

Burge, E. L., and J. M. Roberts. 1993. *Classrooms with a Difference: A Practical Guide to the Use of Conferencing Technologies*. Ontario: University of Toronto Press.

Burnett, G. 2000. Information exchange in virtual communities: A typology. *Inf Res Int Electron J* 5(4). Last retrieved on October 5, 2011 from http://informationr.net/ir/5-4/paper82.html.

Carley, K., L. Ju-Sung, and D. Krackhardt. 2001. Destabilizing networks. *Connections* 24(3):31–4.

Castronova, E. 2001. Virtual worlds: A first-hand account of market and society on the cyberian frontier. CESifo Working Paper Series No. 618. Center for Economic Studies and Ifo Institute for Economic Research, California State University, Fullerton.

Choi, J. H., and J. Danowski. 2002. Cultural communities on the net—Global village or global metropolis? A network analysis of Usenet newsgroups. *J Comput Mediat Commun* 7:3.

Cook, D., and J. Ralston. 2003. Sharpening the Focus: Methodological issues in analyzing on-line conferences. Technology, Pedagogy and Education, 12:3, 361–76.

Cooper, A. 1999. *The Inmates are Running the Asylum*. Indianapolis, IN: SAMS, a division of Macmillan Computer Publishing.

Cyram. 2004. Netminer for Windows. http://netminer.com. Last accessed on October 5, 2011.

December, J. 1997. Notes on defining of computer-mediated communication. *Comput Mediat Commun Mag* 3:1.

Dekker, A. H. 2002. A category-theoretic approach to social network analysis. In *Proceedings of Computing: The Australasian Theory Symposium (CATS) Melbourne*, Australia.

Doran, J., M. Palmer, N. Gilbert, and P. Mellars. 1994. The eos project: Modelling upper paleolithic social change. In *Simulating Societies: The Computer Simulation of Social Phenomena*, ed. G. N. Gilbert and J. Doran. London: UCL Press.

Ducheneaut, N., R. J. Moore, and E. Nickell. 2004. Designing for Sociability in Massively Multiplayer Games: An Examination of the "Third Places" of SWG, Other Players Conference, Denmark.

Electronic Art. 2005. Ultima Online. http://www.uo.com/ (accessed November 8, 2005).

Engestrom, Y. 2001. Expansive learning at work: Toward an activity theoretical reconceptualisation. *J Educ Work* 14:1.

Epstein, J. 2002. Modelling civil violence: An agent-based computational approach. *Proc Natl Acad Sci U S A* 99(3):7243–725.

Epstein, J., and R. Axtell. 1997. *Growing Artificial Societies*. Boston, MA: MIT Press.

Fahy, P. J. 2003. Indicators of support in online interaction. *Int Rev Res Open Distance Learn* 4:1.

Fahy, P. J., G. Crawford, and M. Ally. 2001. Patterns of interaction in a computer conference transcript. *Int Rev Res Open Distance Learn* 2:1.

Ferris, P. 1997. What is CMC? An overview of scholarly definitions. *Comput Mediat Commun Mag* 4:1.

Fortner, R. S. 1993. *International Communication: History, Conflict, and Control of the Global Metropolis*. Belmont, CA: Wadsworth.

Galotti, K. M., B. M. Clinchy, K. Ainsworth, B. Lavin, and A F Mansfield. 1999. A new way of assessing ways of knowing: The Attitudes Towards Thinking and Learning Survey (ATTLS). *Sex Roles* 40(9/10): 745–66.

Garton, L., C. Haythorthwaite, and B. Wellman. 1997. Studying on-line social networks. In *Doing Internet Research*, S. Jones. Thousand Oaks CA: Sage.

Gilbert, N., and K. G. Troitzsch. 2005. *Simulation for the Social Scientist*. Berkshire, U.K.: Open University Press.

Golder, S. A., and J. Donath. 2004. Social roles in electronic communities. Paper presented at the Association of Internet Researchers (AoIR) conference Internet Research 5.0, Brighton, England.

Gunawardena, C., C. Lowe, and T. Anderson. 1997. Analysis of a global online debate and the development of an interaction analysis model for examining social construction of knowledge in computer conferencing. *J Educ Comput Res* 17(4):397–431.

Hanneman, R. A. 2001. Introduction to Social Network Methods. http://faculty.ucr.edu/~hanneman/SOC157/TEXT/TextIndex.html (accessed November 9, 2004).

Haythornthwaite, C., and B. Wellman. 1998. Work, friendship, and media use for information exchange in a networked organisation. *J Am Soc Inf Sci Technol* 49(2):1101–14.

Henri, F. 1992. Computer conferencing and content analysis. In *Collaborative Learning Through Computer Conferencing: The Najaden Papers*, ed. A. R. Kaye, 117–36. Berlin, Germany: Springer-Verlag.

Jager, W., R. Popping, and H. v. d. Sande. 2001. Clustering and fighting in two-party crowds: Simulating the approach-avoidance conflict. *J Artif Societies Soc Simul* 4(3). Last retrieved on October 5, 2011 from http://jasss.soc.surrey.ac.uk/4/3/7.html.

Jones, S. 1995. Computer-mediated communication and community: Introduction. *Comput Mediat Commun Mag* 2:3.

Kelly, R. V. 2004. *Massively Multiplayer Online Role-Playing Games: The People, the Addiction and the Playing Experience*. NC: McFarland & Company.

Klemm, P., M. Hurst, S. L. Dearholt, and S. R. Trone. 1999. Gender differences on internet cancer support groups. *Comput Nurs* 17(2):65–72.

Kolbert, E. 2001. Pimps and Dragons: How an online world survived a social breakdown. The New Yorker, May 28, 2001. http://www.newyorker.com/fact/content/?010528fa_FACT (accessed February 15, 2005).

Kollock, P. 1999. The economies of online cooperation: Gifts and public goods in cyberspace. In *Communities in Cyberspace*, ed. M. A. Smith & P. Kollock. London: Routledge.

Korpela, M., H. A. Soriyan, and K. C. Olufokunbi. 2000. Activity analysis as a method for information systems development. *Scand J Inf Syst* 12:191–210.

Korzenny, F. 1978. A theory of electronic propinquity: Mediated communication in organizations. *Commun Res* 5:3–23.

Krebs, V. 2004. An Introduction to Social Network Analysis. http://www.orgnet.com/sna.html (accessed November 9, 2004).

Kuo, J. 2004. Online video games in mental health. Paper presented at the annual meeting of the American Psychiatry Association, New York.

Kypros-Net Inc. 2005. The World of Cyprus. http://kypros.org. Last accessed on October 5, 2011.

Lamieri, M., and D. Ietri. 2004. Innovation creation and diffusion in a social network: An agent based approach. Last retrieved on October 5, 2011 from http://papers.ssrn.com/sol3/papers .cfm?abstract_id=937255.

Leont'ev, A. N. 1978. *Activity, Consciousness, and Personality*. Upper Saddle River, NJ: Prentice-Hall.

López-Sánchez, M., X. Noria, J. A. Rodríquez-Aguilar, N. Gilbert, and S. Shuster. 2004. Simulation of digital content distribution using a multi-agent simulation approach. In *Recent Advances in Artificial Intelligence Research and Development*, ed. J. Vitria, P. Radeva, and I. Aguilo, 341–8. Amsterdam: IOS Press.

Macy, M. W., and Y. Sato. 2002. Trust, cooperation, and market formation in the U.S. And Japan. Paper presented at the Proceedings of the National Academy of Sciences of the United States of America.

Madden, M., and L. Rainie. 2003. *Pew Internet & American Life Project Surveys*. Washington, DC: Pew Internet & American Life Project.

Mason, R. 1991. Analyzing computer conferencing interactions. *Comput Adult Educ Training* 2(3):161–73.

MaxTsvetovat, and K. Carley. 2002. Knowing the enemy: A simulation of terrorist organizations and counter-terrorism strategies. Paper presented at the CASOS Conference 2002, Pittsburgh, PA.

McLoughlin, C. 1996. A learning conversation: Dynamics, collaboration and learning in computer mediated communication. In *Proceedings of the Third International Interactive Multimedia Symposium*, ed. C. McBeath and R. Atkinson, 267–73. Perth, Western Australia: Promaco Conventions.

McLuhan, M. 1964. *Understanding Media: The Extension of Man*. New York: McGraw-Hill.

Metcalfe, B. 1992. Internet fogies to reminisce and argue at Interop Conference. InfoWorld.

Metz, J. M. 1994. Computer-mediated communication: Literature review of a new context. *IPCT Electron J 21st Century* 2(2):31–49.

Mwanza, D. 2002. *Towards an Activity-Oriented Method for HCI Research and Practice*. Open University. PhD. Milton Keynes, U.K.

NCsoft. 2005. Lineage. http://www.lineage.com/ (accessed November 8, 2005).

Niedner, S., G. Hertel, and S. Hermann. 2000. Motivation in open source projects: An empirical study among linux developers. (accessed April 24, 2008).

NUA Internet Surveys. 2004. http://www.nua.ie/surveys/how_many_online/index.html (accessed October 20, 2004).

Özman, M. 2007. Network formation and strategic firm behaviour to explore and exploit. *J Artif Societies Soc Simul* 11(1):7.

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Pingdom. 2010. Internet 2009 in numbers. http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/. Last accessed on March 10, 2010.

Preece, J. 2000. *Online Communities: Designing Usability, Supporting Sociability*. Chichester, UK: John Wiley and Sons.

Preece, J., and D. Maloney-Krichmar. 2003. Online communities: Focusing on sociability and usability. In *Handbook of Human-Computer Interaction*, ed. J. Jacko and A. Sears, 596–620. Mawhah, NJ: Lawrence Erlbaum Associates Inc. Publishers.

Preece, J., Y. Rogers, and H. Sharp. 2002. *Interaction Design: Beyond Human-Computer Interaction*. New York: John Wiley & Sons.

Raymond, E. 1999. *The Cathedral and the Bazaar: Musings on Linux and Open Source from an Accidental Revolutionary*. Sebastapol, CA: O'Reilly and Associates.

Rheingold, H. 1993. *The Virtual Community: Homesteading on the Electonic Frontier*. Reading: Addison-Wesley.

Rice, R. 1994. Network analysis and computer mediated communication systems. In *Advances in Social Network Analysis*, ed. S. W. J. Galaskiewkz. Newbury Park, CA: Sage.

Rice, R. E., A. E. Grant, J. Schmitz, and J. Torobin. 1990. Individual and network influences on the adoption and perceived outcomes of electronic messaging. *Soc Netw* 12:17–55.

Savicki, V., D. Lingenfelter, and D. Kelley. 1996. Gender language style and group composition in internet discussion groups. *J Comput Mediated Commun* 2(3). http://jcmc.indiana.edu/vol2/issue3/savicki.html.

SCOTCIT. 2003. Enabling large-scale institutional implementation of communications and information technology (ELICIT). Using Computer Mediated Conferencing. http://www.elicit.scotcit.ac.uk/modules/cmc1/welcome.htm (accessed November 2, 2004).

Scott, J. 2000. *Social Network Analysis: A Handbook*. 2nd ed. London: Sage.

Sony. 2005. Star Wars Galaxies. http://starwarsgalaxies.station.sony.com/ (accessed November 8, 2005).

Squire, K., and C. Steinkeuhler. In press. Generating CyberCulture/s: The case of star wars galaxies. In *Cyberlines: Languages and Cultures of the Internet,* ed. D. Gibbs and K. L. Krause, 2nd ed. Albert Park, Australia: James Nicholas Publishers.

Suler, J. 2004. The Final Showdown Between In-Person and Cyberspace Relationships. http://www1.rider.edu/~suler/psycyber/showdown.html (accessed November 3, 2004).

Taylor, T. L. 2002. "Whose Game Is This Anyway?": Negotiating Corporate Ownership in a Virtual World." CGDC Conference Proceedings, 227–42. Tampere, Finland.

Turner, T. C., M. A. Smith, D. Fisher, and H. T. Welser. 2005. Picturing usenet: Mapping computer-mediated collective action. *J Comput Mediated Commun* 10(4), Article 7.

U.S Census Bureau. 2004. Global Population Profile 2002, http://www.census.gov/ipc/www/wp02.html (accessed October 20, 2004).

Usability Net. 2003. UsabilityNet. http://www.usabilitynet.org/ (accessed December 3, 2004).

Vaknin, S. 2002. TrendSiters: Games People Play. Electronic Book Web. http://12.108.175.91/ebookweb (accessed February 23, 2002).

Vygotsky, L. 1930. *Mind and Society*. Cambridge, MA: Harvard University Press.

Waldrop, M. 1992. *Complexity: The Emerging Science at the Edge of Chaos*. New York: Simon and Schuster.

Wallace, P. 1999. *The Psychology of the Internet*. Cambridge: Cambridge University Press.

Wellman, B. 1982. Studying personal communities. In *Social Structure and Network Analysis*, ed. P. M. N. Lin. Beverly Hills, CA: Sage.

28 Virtual Environments

Kay M. Stanney and Joseph V. Cohn

CONTENTS

28.1 INTRODUCTION

Interactive computing has evolved through the years from cryptic command-based interfaces to a collection of task-based applications to ecologically valid immersive environments, each advance dissolving more of the barrier between users and their desired actions. To many, with virtual environments (VEs), we have reached the panacea in interactive computing. Such environments immerse users in realistic settings, allowing them to engage in an intuitive and intimate manner with their digital universe. Such capability affords the opportunity to "learn by doing," "train like we fight," and "involve me and I understand." Although early VEs were low in resolution and sluggish in responsiveness, advances in display and tracking technology have largely resolved these issues such that today VE applications abound. However, considerable human–computer interaction (HCI) research and development are required to resolve lingering issues, such as cybersickness and usability, if VE technology is to be openly embraced by users. This chapter reviews the current state of the art in VE technology, provides design and implementation strategies, discusses health and safety concerns and potential countermeasures, and presents the latest in VE usability engineering approaches. Current efforts in a number of application domains are reviewed. This chapter should enable readers to better specify design and implementation requirements for VE applications and prepare them to use this advancing technology in a manner that minimizes health and safety concerns.

28.2 SYSTEM REQUIREMENTS

A VE is a computer-generated immersive environment that can simulate both real and imaginary worlds, often times in three dimensions (3D). Current VE applications are primarily intriguing visual and auditory experiences, with a smaller number incorporating additional sensory modalities, such as haptics and smell. These worlds are driven by hardware, which provides the hosting platform and multimodal presentation, allows for physical interaction, and tracks the whereabouts of users as they traverse the virtual world and software to model and generate the virtual world and their autonomous agents and support communication networks that link multiple users (see Figure 28.1).

More specifically, hardware interfaces consist primarily of the following:

- Interface devices used to present multimodal information and sense the VE
- Tracking devices used to identify head and limb position and orientation
- Interaction techniques that allow users to navigate through and interact with the virtual world

Software interfaces include the following:

- Modeling software used to generate VEs
- Autonomous agents that inhabit VEs
- Communication networks used to support multiuser VEs

28.2.1 Hardware Requirements

VEs require very large physical memories, high-speed processors, high-bandwidth mass-storage capacity, and high-speed interface ports for interaction devices (Durlach and Mavor 1995). These requirements are easily met by today’s high-speed, high-bandwidth computing systems, many of which have surpassed the Gigahertz barrier. The future looks even brighter with promises of massive parallelism in multicore and many-core processor architectures (Holmes, Williams, and Tilke 2010), which will allow tomorrow’s computing systems to be exponentially faster than their ancestors. With the rapidly advancing ability to generate complex and large-scale virtual worlds, hardware advances in multimodal input/output (I/O) devices, tracking systems, and interaction techniques are needed to support generation of increasingly engaging virtual worlds. In addition, the coupling of augmented cognition and VE technologies can lead to substantial gains in the ability to evaluate their effectiveness.

28.2.1.1 Multimodal Inputs/Outputs

To present a multimodal VE (see Chapter 18), multiple devices are used to present information to VE users. In terms of VE projection systems, the one that has received the greatest attention, both in hype and disdain, is almost certainly the head-mounted display (HMD). One benefit of HMDs is their compact size, as an HMD when coupled with a head tracker can be used to provide a similar visual experience as a multitude of bulky displays such as those associated with spatially immersive displays (SIDs) and desktop solutions. In addition, HMDs are suggested to enhance situation awareness (SA), enable correct decision making, and reduce workload by allowing users to turn their heads and eyes to fully perceive the environment, decreasing multimodal clutter, providing an intuitive means of presenting spatialized multimodal warnings and alerts, and redundantly coding critical cues (e.g., external threats, navigational waypoints),

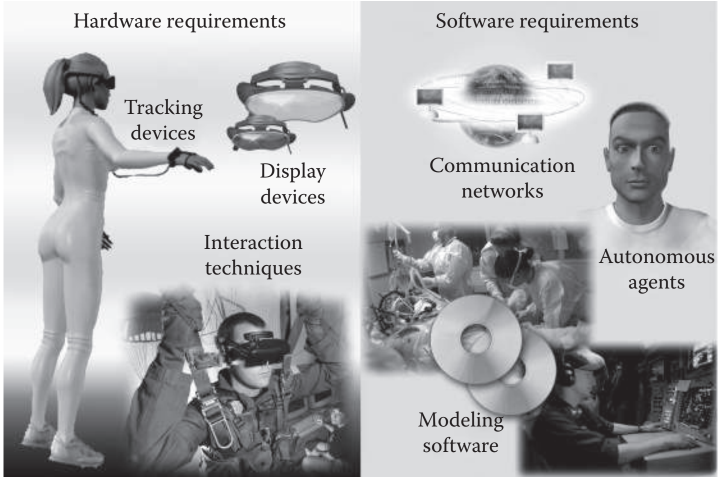


FIGURE 28.1 Hardware and software requirements for virtual environment generation.

for example, by using audio cues to direct visual attention (Melzer and Rash 2009).

There are three main types of HMDs: (1) monocular (e.g., one image source is viewed by a single eye), (2) biocular (e.g., one image source viewed by both eyes), and (3) binocular (e.g., stereoscopic viewing via two image generators, with each eye viewing an independent image source) (Melzer et al. [2009]). A monocular HMD design is best when projecting moving maps or text information that must be read on the move (e.g., dismounted Warfighter) or to allow viewing of imagery with the simplest, lightest (in terms of head-supported weight/mass), and least costly (both monetarily and in terms of power consumption) solution. The downside of monocular displays is that they have a small field-of-view (FOV), convey no stereoscopic depth information, have the potential for a laterally asymmetric center-of-mass (CM), and may have issues associated with focus, eye dominance, binocular rivalry, and ocular-motor instability. For a wide FOV, more effective target recognition, and a more comfortable viewing experience, a biocular or binocular solution is needed. Biocular solutions present no interocular rivalry and are lighter, easier to adjust, and less expensive than binocular solutions. The disadvantages of biocular displays are that they are heavier, more complex to align, focus, and adjust, and have reduced luminance when compared with monocular displays. Binocular displays can present stereo viewing (via field-sequential single-screen displays with shutter glasses, single-screen polarized displays, or dual-screen HMDs), which provides for better depth information than monocular and biocular solutions and have a symmetrical CM. On the downside, binocular solutions are heavy, require more complex alignment, focus, and adjustments than monocular, and are expensive. Biocular and binocular solutions are particularly well suited when creating fully immersive VEs for gaming or training systems, as their large FOV provides a more compelling sense of immersion.

When coupled with tracking devices, HMDs can be used to present 3D visual scenes that are updated as a user moves his or her head about a virtual world. Although this often provides an engaging experience, due to poor optics, sensorial mismatches, and slow update rates, these devices are also often associated with adverse effects such as eyestrain and nausea (Stanney and Kennedy 2008). In addition, although HMDs have come down substantially in weight, rendering them more suitable for extended wear, they are still hindered by cumbersome designs, obstructive tethers, suboptimal resolution, and insufficient fields of view. These shortcomings may be the reason behind why, in a review of HMD devices, approximately a third had been discontinued by their manufacturers (Bungert 2007). Nevertheless, of the HMDs available, there are several low- to mid-cost models, which are relatively lightweight and provide a horizontal FOV and resolution far exceeding predecessor systems.

Low-technology stereo viewing VE display options include anaglyph methods, where a viewer wears glasses with distinct color polarized filters, usually with the left-image data placed in the red channel of an electronic display and the right-image data in the blue channel; parallel or cross-eyed methods, in which right and left images are displayed adjacently (parallel or crossed), requiring the viewer to actively fuse the separate images into one stereo image; parallax barrier displays, in which an image is made by interleaving columns of two images from a left- and right-eye perspective image of a 3D scene; polarization methods, in which the images for the left and right eyes are projected on a plane through two orthogonal linearly polarizing filters (e.g., the right image is polarized horizontally; the left is polarized vertically) and glasses with polarization filters are donned to see the 3D effect; Pulfrich methods, in which an image of a scene moves sideways across the viewer's FOV and one eye is covered by a dark filter so that the darkened image reaches the brain later causing stereo disparity; and shutter glass methods in which images for the right and left eyes are displayed in quick alternating sequence and special shutter glasses are worn that "close" the right or left eye at the correct time (Konrad and Halle 2007; Vince 2004). All these low-technology solutions are limited in terms of their resolution, the maximum number of views that they can display, and clunky implementation; they can also be associated with pseudoscopic images (e.g., the depth of an object can appear to flip inside out).

Other options in visual displays include SIDs (e.g., displays that surround viewers physically with panoramic large FOV imagery generally projected via fixed front or rear projection display units; Konrad and Halle [2007]; Majumder [2003]), desktop stereo displays, and volumetric displays that fill a volume of space with a "floating" image (Konrad and Halle 2007). Examples of SIDs include the Cave Automated VE (CAVE) (Cruz-Neira, Sandin, and DeFanti 1993), PowerWall and Infinity Wall (Majumder, He, Towles, and Welch, 2000). Issues with SIDs include a stereo view that is correct for only one or a few viewers, noticeable overlaps between adjacent projections, and image warp on curved screens. Blue-c addresses some of these concerns by combining simultaneous acquisition of multiple 3D video streams with advanced 3D projection technology (Gross et al. 2003).

Desktop display systems have advantages over SIDs because they are smaller, easier to configure in terms of mounting cameras and microphones, easier to integrate with gesture and haptic devices, and more readily provide access to conventional interaction devices, such as mice, joysticks, and keyboards. Issues with such displays include stereo that is only accurate for one viewer and a limited display volume.

Volumetric displays provide visual accommodation depth cues and vertical parallax, which are particularly useful for scenes that require viewing from a multitude of viewing angles, generally without the need for goggles; however, they do not maintain accurate occlusion cues (often considered the strongest depth cues) for all viewers (Konrad and Halle 2007). Perspecta is an example of a swept-volume display that uses a flat, double-sided screen with a rotating projected image to sweep out a hemispherical image volume (Favalora 2005). DepthCube is an example of a static-volume display that uses electronically addressable elements (i.e., a digital micromirror device imaging system) to scan out the image volume (Sullivan 2004). Issues with volumetric displays include low resolution and the tendency for transparent images to lose interposition cues. Also, view-independent shading of objects is not possible with volumetric displays, and current solutions do not exhibit arbitrary occlusion by interposition of objects (Konrad and Halle 2007).

The way of the future seems to be direct virtual retinal displays, where images are projected directly onto the human retina with a low-energy laser or LCDs (McQuaide et al. 2003), as well as displays that represent the physical world around us, such as autostereoscopic omni-directional light-field displays, which present interactive 3D graphics to multiple simultaneous viewers 360° around the display (Jones et al. 2007). If designed effectively, these next-generation devices should eliminate the tethers and awkwardness of current designs while enlarging the FOV and enhancing resolution.

When VEs provide audio (see Chapter 10), the interactive experience is generally greatly enhanced (Shilling and Shinn-Cunningham 2002). Audio can be presented via spatialized or nonspatialized displays. Just as stereo visual displays are a defining factor for VE systems, so are "interactive" spatialized audio displays (e.g., those with "on-the-fly" positioning of sounds). VRSonic's SoundScape3D (http://www.vrsonic.com/), Firelight's FMod (http://www.fmod.org/), and AuSIM3D (http://ausim3d.com/) are examples of positional 3D audio technology. There have been promising developments in new sound modeling paradigms (e.g., VRSonic's ViBe technology) and sound design principles that will hopefully lead to a new generation of tools for designing effective spatial audio environments (Fouad 2004; Fouad and Ballas 2000; Jones, Stanney, and Fouad 2005).

Developers must decide if sounds should be presented via headphones or speakers. For nonspatialized audio, most audio characteristics (e.g., timbre, relative volume) are generally considered to be equivalent, whether projected via headphones or speakers. This is not so for spatialized audio, in which the presentation technique impacts how audio is rendered for the display and presents the developer with important design choices.

While in the past, headphone spatialization required expensive, specialized hardware to achieve real-time rates, modern multicore processors and the availability of powerful graphics processing units (GPU) have made it possible to render complex audio environments over headphones using general-purpose computers. With binaural rendering, a sound can be placed in any location, right or left, up or down, near or far, via the use of a head-related transfer function (HRTF) to represent the manner in which sound sources change as a listener moves his or her head (Begault 1994; Butler 1987; Cohen 1992). However, for optimal results, the HRTFs used for rendering must be personalized for each individual user. One method of doing this is to actually measure each user's HRTF for use in rendering. This approach generally involves a fairly lengthy measurement procedure using specialized hardware. Recently, there have been efforts to develop fast and low-cost approaches to HRTF measurements (Zotkin et al. 2006), which may, in the future, make personalized HRTF rendering practical for general use. An alternative approach to measure HRTFs is to use a best-fit HRTF selection process in which one finds the nearest matching HRTF in a database of candidate HRTFs by either comparing the physiological characteristics of stored HRTFs with those of a target user (Algazi et al. 2001) or by using a subjective selection process to find the best-fit HRTF (Seeber and Fastl 2003). Another consideration that should be taken into account when choosing headphone rendering is that, for immersive displays, head trackers must be used to achieve proper relative positioning of sound sources. Also, rendering spatial audio for groups of users over headphones may not be practical for more than a few users.

An alternative approach to headphone spatialization is the use of loudspeaker arrays (Ballas et al. 2001). Loudspeaker arrays can range in size from relatively small surround sound configurations with 2, 4, 5, 7, or 10 loudspeakers up to hundreds of loudspeakers. The differentiating factors among loudspeaker arrays are the speaker layout, number of loudspeakers comprising the array, and algorithms used to render spatial audio. Generally speaking, increasing the number of loudspeakers in the array results in more accurate spatialization. The manner in which loudspeakers are laid out in a listening area is closely related to the size of the array. Planar loudspeaker configurations require a smaller number of loudspeakers, but are only capable of creating a 2D sound field. Volumetric configurations, on the other hand, can create a 3D sound field, but require a larger number of loudspeakers and a more elaborate setup. Recently, VRSonic introduced a spherical loudspeaker array system called the AcoustiCurve. It provides a volumetric array in a spherical configuration around the listening space (see Figure 28.2).

The rendering algorithm used for spatialization is also closely tied to the loudspeaker array size and configuration. Pair-wise panning algorithms are the simplest form of spatialization and create a positional sound source by manipulating the amplitude of the signal arriving at two adjacent loudspeakers in the array (Mouba 2009). An extension to this idea is vector based amplitude panning, where the source is panned among three loudspeakers forming a triangle in a volumetric array (Pulkki 1997). Another spatialization algorithm that is gaining popularity is wave field synthesis (WFS); a technique based on Huygens' Principle (Spors and Ahrens 2010). The WFS technique creates a positional source within the listening space by recreating the incident wavefront of a virtual source using a loudspeaker array. The advantage of WFS is that it does not suffer from the "sweet spot" problem, and so listeners can get an accurate impression of the synthesized sound field at any location within the listening space; this is not the case with pair-wise panning (Shilling and Shinn-Cunningham 2002). The primary drawback of WFS is that it requires a large number of loudspeakers and considerable processing power to recreate the incident wavefront.

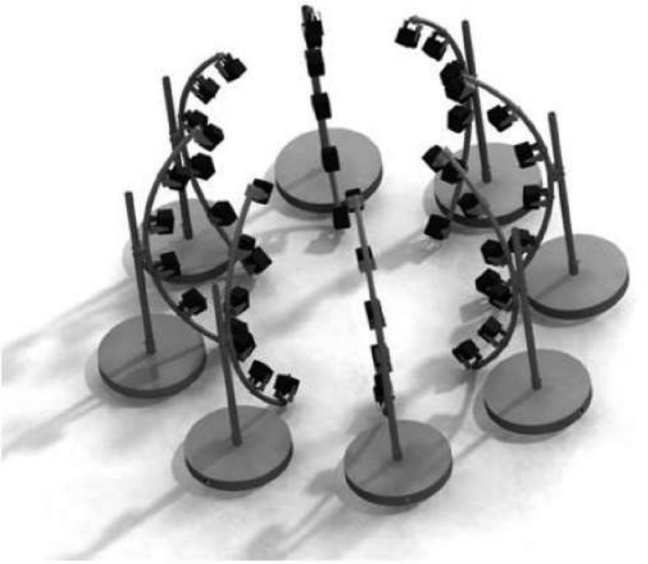


FIGURE 28.2 Volumetric speaker array

Whether using headphones or loudspeaker arrays, spatialization is only one component of simulating a sound field, and developers should carefully consider the level of fidelity required by the application when choosing an audio rendering system. Properly synthesizing a virtual soundscape requires modeling the full propagation path of sound, including source model, spreading loss, air absorption, material absorption, and material reflection. Accurately modeling the full propagation path in real time is beyond the capabilities of current computers. However, there is promising research in the use of GPU processors to achieve real-time rates using ray casting methods (Jedrzejewski 2004).

Although not as commonly incorporated into VEs as visual and auditory interfaces, haptic devices (see Chapter 9) can be used to enhance aspects of touch and movement of the hand or body segments while interacting with a VE. Haptic devices have been classified as passive (unidirectional; e.g., keyboard, mouse, trackball) versus active (bidirectional, thereby supporting two-way communications between human and interactive system; Hale and Stanney [2004]; e.g., force-reflecting robotic arm), grounded (e.g., joystick) versus ungrounded (e.g., exoskeleton-type haptic devices), net-force (e.g., PHANTOM device or textured surfaces) versus tactile devices (e.g., tactile pin arrays), and impedance control (i.e., user's input motion is measured and an output force is returned) versus admittance control (e.g., user's input forces are measured and motion is fed back to the user) (Basdogan and Loftin 2008). In general, haptic displays are effective at alerting people to critical tasks (e.g., warning), providing a spatial frame of reference within one's personal space, and supporting hand–eye coordination tasks. Texture cues, such as those conveyed via vibrations or varying pressures, are effective as simple alerts and may speed reaction time and aid performance in degraded visual conditions (Akamatsu 1994; Biggs and Srinivasan 2002; Massimino and Sheridan 1993; Mulgund et al. 2002). Kinesthetic devices are advantageous when tasks involve hand–eye coordination (e.g., object manipulation), where haptic sensing and feedback are key to performance. Currently available haptic-interaction devices include static displays (e.g., convey deformability or Braille); vibrotactile, electrotactile, and pneumatic displays (e.g., convey tactile sensations such as surface texture and geometry, surface slip, surface temperature); force feedback systems (e.g., convey object position and movement distances); and exoskeleton systems (e.g., enhance object interaction and weight discrimination) (Hale and Stanney 2004). Minamizawa et al. (2008) suggest that to provide natural haptic feedback, such interfaces should be bimanual and wearable and aim to enhance the existence and operability of virtual objects while not disturbing the motion and behavior of users. Currently, there are several wearable haptic displays that can be used in VEs, such as CyberGlove Systems' CyberGlove, CyberTouch, CyberGrasp, and CyberForce (http://www.cyberglovesystems.com/) and Immerz's KOR-fx (Kinetic Omni-directional Resonance effect) acousto-haptic technology, the latter of which translates the audio signals from an interactive environment into vibrations that can be felt throughout the body and experienced as the sensation of rain, wind, weight shift, and G-forces (www.Immerz.com). Beyond supporting hand–eye coordination tasks and conveying simple alerts, haptic can be used to communicate a grammar through structured strings of tactile symbols (Fuchs et al. 2008). Such a tactile language has been used at a concept level to support urban military operations, specifically in support of unit coordination and room clearing tasks (Johnston, Hale and Axelsson 2010). Beyond communicating a command-based vocabulary, haptics can also be used to provide exteroceptive feedback, for example, by presenting tactile cues to enhance SA or optimize human performance. It has been suggested that such a solution could more closely couple operators with unmanned aerial systems (Johnston et al. 2010). The future may bring volumetric haptic displays, which project a touch-based representation of a surface onto a 3D volumetric space and allow users to feel the projected surface with their hands (Acosta and Liu 2007) through haptic-rendering techniques (Basdogan et al. 2008), tearables that allow users to experience the real sense of tearing paper (Maekawa et al. 2009), and other such interactive tactile solutions.

The "vestibular system can be exploited to create, prevent, or modify acceleration perceptions" in VEs (Lawson, Sides, and Hickinbotham 2002, p. 137). For example, by simulating acceleration cues, a person can be psychologically transported from his or her veridical location, such as sitting in a chair in front of a computer, to a simulated location, such as the cockpit of a moving airplane. Although vestibular cues can be stimulated via many different techniques in VEs, three of the most promising methods are physical motion of the user (e.g., motion platforms), wide FOV visual displays that induce vection (e.g., an illusion of self-motion), and locomotion devices that induce illusions of self-motion without physical displacement of the user through space (e.g., walking in place, treadmills, pedaling, foot platforms) (Hettinger 2002; Hollerbach 2002; Lawson, Sides, and Hickinbotham 2002). Of these options, motion platforms are probably the most advanced. For example, Sterling, Magee, and Wallace (2000) integrated a small motion-based platform with a VE designed for helicopter landing training and found it to be comparable with a high-cost, large-scale helicopter simulator in terms of training effectiveness. Motion platforms are generally characterized via their range of motion/degrees of freedom (DOF) and actuator type (Isdale 2000). In terms of range of motion, motion platforms can move a person in many combinations of translational (e.g., surge-longitudinal motion, sway-lateral motion, heave-vertical motion) and rotational (e.g., roll, pitch, yaw) DOF. A single-DOF translational motion system might provide a vibration sensation via a "seat shaker." A common 6 DOF configuration is a hexapod, which consists of a frame with six or more extendable struts (actuators) connecting a fixed base to a movable platform. In terms of actuators, electrical actuators are quiet and relatively maintenance free; however, they are not very responsive and they cannot hold the same load as can hydraulic or pneumatic systems. Hydraulic and pneumatic systems are smoother, stronger, and more accurate; however, they require compressors, which may be noisy. Servos are expensive and difficult to program.

Olfaction could be added to VE systems to stimulate emotion or enhance recall (Basdogan and Loftin 2008). There have been several efforts made to support advances in olfactory interaction (Gutierrez-Osuna 2004; Jones et al. 2004; Washburn and Jones 2004; Washburn et al. 2003). One example of an olfactory system is the Scent Pallet (http://www.enviroscent.com/), which is a computer peripheral, USB device that uses up to 8 scent cartridges, fans, and an air compressor to deliver different types of scents. This system has been incorporated into the Full Spectrum Virtual Iraq/Afghanistan PTSD Therapy Application to provide the smell of rubber, cordite, garbage, body odor, smoke, diesel fuel, gunpowder, and other scents of the battlefield (Rizzo, Rothbaum, and Graap 2006). These scents can be used as direct stimuli (e.g., scent of burning rubber) or as general cues to increase immersion (e.g., ethnic food cooking). The Scent Pallet was used to present vanilla, pizza, coffee, whiskey, beer, brandy, tequila, gin, scotch, red wine, white wine, cigarette smoke, and pine tree scents in an alcohol cue reactivity assessment system, which was found to be highly effective in stimulating subjective alcohol cravings (Bordnick et al. 2008). Although several have mentioned the incorporation of gustatory stimulation, there are currently no functioning systems (Basdogan and Loftin 2008).

28.2.1.2 Tracking Systems

Tracking systems allow determination of users' head or limb position and orientation or the location of hand-held devices to allow interaction with virtual objects and traversal through 3D computer-generated worlds (Foxlin 2002). Tracking is what allows the visual scene in a VE to coincide with a user's point of view, thereby providing an egocentric real-time perspective. Tracking systems must be carefully coupled with the visual scene, however, to avoid unacceptable lags (Kalawsky 1993). Advances in tracking technology have been realized in terms of drift-corrected gyroscopic orientation trackers, outside-in optical tracking for motion capture, and laser scanners (Foxlin 2002). The future of tracking technology is likely hybrid tracking systems (http://www.intersense.com/hybrid_technology.aspx), such as optical-inertial, GPS-inertial, magnetic-inertial, digital acoustic-inertial, and optical-magnetic hybrid solutions.

Tracking technology also allows for gesture recognition, in which human position and movement are tracked and interpreted to recognize semantically meaningful gestures (Turk 2002). Gestures can be used to specify and control objects of interest, direct navigation, manipulate the environment, and issue meaningful commands. Gesture tracking devices that are worn (e.g., gloves, bodysuits) are currently more advanced than passive techniques (e.g., computer vision), yet the latter hold much promise for the future, as they can provide more natural, noncontact, and less obtrusive solutions than those that must be worn; yet limitations need to be overcome in terms of accuracy, processing speed, and generality (Erol et al. 2007).

28.2.1.3 Interaction Techniques

Although one may think of joysticks and gloves when considering VE interaction devices, there are many techniques that can be used to support interaction with and traversal through a VE. Interaction devices support traversal, pointing, and selection of virtual objects, tool usage (e.g., through force and torque feedback), tactile interaction (e.g., through haptic devices), and environmental stimuli (e.g., temperature, humidity) (Bullinger, Breining, and Braun 2001).

Supporting traversal throughout a VE, via motion interfaces, is of primary importance (Hollerbach 2002). Motion interfaces are categorized as either active (e.g., locomotion) or passive (e.g., transportation). Active motion interfaces require self-propulsion to move about a VE (e.g., treadmill, pedaling device, foot platforms). Passive motion interfaces transport users within a VE without significant user exertion (e.g., inertial motion, as in a flight simulator, or noninertial motion, such as in the use of a joystick or gloves). The utility, functionality, cost, and safety of locomotion interfaces beyond traditional options (e.g., joysticks) have yet to be proven. In addition, beyond physical training, concrete applications for active motion interfaces have yet to be clearly delineated. There are, however, some example applications, such as Arch-explore, which is a real walking user interface that adapts redirected walking to allow exploration of large-scale virtual models of architectural scenes in a room-sized VE (Bruder, Steinicke, and Hinrichs 2009).

Another interaction option is speech control (see Chapter 16). Continuous speech recognition systems are currently under development, such as Parakeet (Vertanen and Kristensson 2009), PocketSphinx (Huggins-Daines et al. 2006), and PocketSUMMIT (Hetherington 2007). For these systems to provide effective interaction, however, additional advances are needed in acoustic and language-modeling algorithms to improve the accuracy, usability, and efficiency of spoken language understanding; such systems are still a ways away from offering conversational speech.

To support natural and intuitive interaction, a variety of interaction techniques can be coupled. For example, combining speech interaction with nonverbal gestures and motion interfaces can provide a means of interaction that closely captures real-world communications.

28.2.1.4 Augmented Cognition Techniques

Augmented cognition is an emerging computing paradigm in which users and computers are tightly coupled via physiological gauges that measure the cognitive state of users and adapt interaction to optimize human performance (Stanney et al. 2009). If incorporated into VE applications, augmented cognition could provide a means of evaluating their engaging and compelling nature. For example, neuroscience studies have established that differential aspects of the brain are engaged when learning different types of materials, and the areas in the brain that are activated change with increasing competence (Carroll et al. 2010; Kennedy et al. 2005). Thus, if VE users were immersed in an educational experience, augmented cognition technology could be used to gauge if targeted areas of the brain were being activated and dynamically modify the content of a VE learning curriculum if desired activation patterns were not being generated. Physiological measures could also be used to detect the onset of cybersickness (see Section 28.4.1) and to assess the engagement, awareness, and anxiety of VE users, thereby potentially providing much more robust measures of immersion and presence (see Section 28.5.2). Such techniques could prove invaluable to entertainment VE applications (cf., Badiqué et al. [2002]) that seek to provide the ultimate experience, military training VE applications (cf., Knerr et al. [2002]) that seek to emulate the "violence of action" found during combat, medical training applications (Wiecha et al. 2010) that seek to enhance traditional laboratory-based and classroom training practices, and therapeutic VE applications (cf., North, North, and Coble 2002; Strickland et al. [1997]) that seek to overcome disorders such as fear of heights or flying.

28.2.2 Software Requirements

Software development of VE systems has progressed tremendously, from proprietary and arcane systems to development kits that run on multiple platforms (e.g., general-purpose operating systems to workstations). VE system components have become modular and distributed, thereby allowing VE databases (e.g., editors used to design, build, and maintain virtual worlds) to run independently of visualizers and other multimodal interfaces via network links. Standard application program interfaces (APIs; e.g., OpenGL, Open Inventor, Direct3D, Mesa3D) allow multimodal components to be hardware-independent. VE programming languages are maturing, with APIs, libraries (OpenGL Performer), and scripting languages (e.g., JavaScript, Lua, Linden, Mono, Perl, Python, Ruby) allowing nonprogrammers to develop virtual worlds (Stanney and Zyda 2002). Advances are also being made in modeling of autonomous agents and communication networks used to support multiuser VEs.

28.2.2.1 Modeling

A VE consists of a set of geometry, the spatial relationships between the geometry and the user, and the change in geometry invoked by user actions or the passage of time (Kessler 2002). Generally, modeling starts with building the geometry components (e.g., graphical objects, sensors, viewpoints, and animation sequences) (Kalawsky 1993). These are often converted from CAD data. These components then get imported into the VE modeling environment and rendered when appropriate sensors are triggered. Color, surface textures, and behaviors are applied during rendering. Programmers control the events in a VE by writing task functions, which become associated with the imported components.

A number of 3D modeling languages and toolkits are available that provide intuitive interfaces and run on multiple platforms and renderers (e.g., 3D Studio Max, AC3D, ZBrush, modo 401, Nexus, AccuRender, 3d ACIS Modeler, Ashlar-Vellum's Argon/Xenon/Cobalt, Carrara, CINEMA 4D, DX Studio, EON Studio, solidThinking) (Ultimate 3D Links 2010). In addition, there are scene management engines (e.g., OpenSceneGraph, NVIDIA's SceniX) and game engines (e.g., Real Virtuality) that allow programmers to work at a higher level, defining characteristics and behaviors for more holistic concepts (Karim et al. 2003; Menzies 2002). There have also been advances in photorealistic rendering tools (e.g., EI Technology's Amorphium), which are evolving toward full-featured, physics-based global illumination rendering systems (e.g., RenderPark). Taken together, these advances in software modeling allow for the generation of complex and realistic VEs that can run on a variety of platforms, permitting access to VE applications by both small- and large-scale application-development budgets.

28.2.2.2 Autonomous Agents

Autonomous agents are synthetic or virtual human entities that possess some degree of autonomy, social ability, reactivity, and proactiveness (Allbeck and Badler 2002). There are several types of agents (Serenko and Detlor 2004), including user agents (i.e., assist users by interacting with them, knowing their preferences and interests, and acting on their behalf), service agents (i.e., seamlessly collaborate with different parts of a system and perform more general tasks in the background, unbeknownst to users), embedded agents (i.e., interact with user and system to hide task complexity and make the overall user experience more exciting and enjoyable), and stand-alone agents (i.e., employ leading-edge technologies and lay down the foundation for new architectures, standards, and innovative formats of agent-based computing). Autonomous agents can have many forms (e.g., human, animal), which are rendered at various levels of detail and style, from cartoonish to physiologically accurate models; the form of the agent has been found to influence behavior both during and post VE exposure (i.e., the Proteus Effect, where people infer their expected behaviors and attitudes from observing the appearance of their avatar; Yee, Bailenson, and Ducheneaut [2009]). Such agents are a key component of many VE applications involving interaction with other entities, such as adversaries, instructors, or partners (Stanney and Zyda 2002). Considerable work is being done to enhance the believability of such agents. For example, Heylen et al. (2008) found that when human-like eye gaze behavior was incorporated into agents, users communicated with such agents more effectively, and of utmost importance, human performance was also found to be enhanced with the more life-like agents. As our understanding of how best to design autonomous agents evolves, such principles will be important to incorporate into their design to enhance the overall engagement and effectiveness of virtual worlds.

There has been significant research and development in modeling embodied autonomous agents. As with object geometry, agents are generally modeled off-line and then rendered during real-time interaction. Although the required level of detail varies, modeling of hair and skin adds realism to an agent's appearance (Allbeck and Badler 2002). There are a few toolkits available to support agent development, with one of the most notable offered by Boston Dynamics, Inc. (BDI) (http://www.bostondynamics.com/bd_diguy.html/), a spin-off from the MIT Artificial Intelligence Laboratory. BDI's DI-Guy allows VE developers to quickly integrate humans into their VEs, providing artificial intelligence to the characters, thereby enabling agents to autonomously navigate and react to their changing environment. Another option is ArchVision's 3D Rich Photorealistic Content (RPC) People (http://www.archvision.com/RPCPeople.cfm).

28.2.2.3 Networks

Distributed networks allow multiple users at diverse locations to interact within the same VE. Improvements in communication networks are required to allow realization of such shared experiences in which users, objects, processes, and autonomous agents from diverse locations interactively collaborate (Durlach and Mavor 1995). Yet, the foundation for such collaboration has been built within Internet2 (http://www.internet2.edu/), a next-generation Internet Protocol (IP) that delivers production network services for research and education institutions. This optical network could meet the high-performance demands of VEs, as it allows user-based allocation of high-capacity data circuits over a fiber-optic network. In addition, the Large Scale Networking (LSN) Coordinating Group (http://www.nitrd.gov/subcommittee/lsn.aspx) aims to develop leading-edge networking technologies and services, including programs in network security, new network architectures, heterogeneous networking (optical, mobile wireless, sensornet, etc.), federation across networking domains, grid and collaboration networking tools and services, with a goal of assuring that the next generation of the Internet will be scalable, trustworthy, and flexible. There are additional novel network technologies including IP multicasting (i.e., a routing technique to prioritize one-to-many communication over an IP infrastructure in a network), quality of service (i.e., resource reservation control mechanisms), and IPv6 (i.e., also called IPng [or IP Next Generation], a next-generation IP addressing system) that could support distributed VE applications, which can leverage the special capabilities (e.g., high-bandwidth, low-latency, and low-jitter) of these advancing network technologies to provide shared virtual worlds.

28.3 DESIGN AND IMPLEMENTATION STRATEGIES

Although many conventional HCI techniques can be used to design and implement VE systems, there are unique cognitive, content, product liability, and usage protocol considerations that must be addressed (see Figure 28.3).

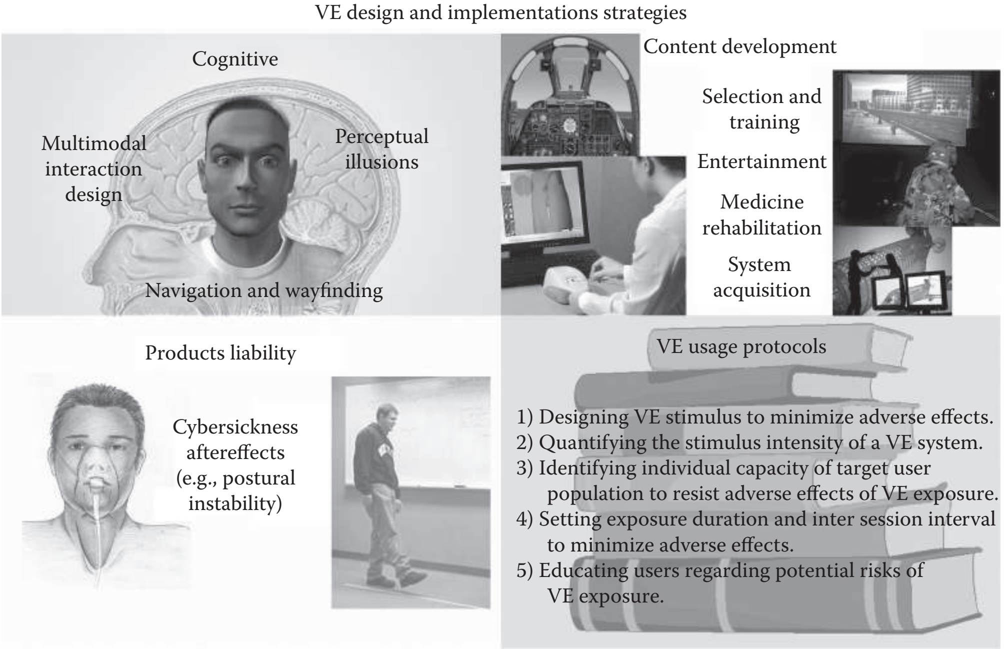

VE design and implementations strategies
Cognitive
Multimodal interaction design
Perceptual illusions
Navigation and wayfinding
Content development
Selection and training
Entertainment
Medicine rehabilitation
System acquisition
Products liability
Cybersickness aftereffects (e.g., postural instability)
VE usage protocols
1) Designing VE stimulus to minimize adverse effects.
2) Quantifying the stimulus intensity of a VE system.
3) Identifying individual capacity of target user population to resist adverse effects of VE exposure.
4) Setting exposure duration and inter session interval to minimize adverse effects.
5) Educating users regarding potential risks of VE exposure.

FIGURE 28.3 Design and implementation considerations for virtual environments.

28.3.1 Cognitive Aspects

The fundamental objective of VE systems is to provide multimodal interaction or, when sensory modalities are missing, perceptual illusions that support human information processing in pursuit of a VE application's goals, which could range from training to entertainment. Ancillary yet fundamental to this goal is to minimize cognitive obstacles, such as navigational difficulties, that could render a VE application's goals inaccessible.

28.3.1.1 Multimodal Interaction Design

VEs are designed to provide users with immersive experiences that allow for direct manipulative and intuitive interaction with multisensory stimulation (Bullinger et al. 2001). The goals of providing this multimodal interaction within a VE are to achieve human–system interaction that is as natural as possible and to increase the robustness of this interaction by using redundant or complementary cues (Reeves et al. 2004). If designed effectively, engagement in such immersive multimodal VE experiences can lead to high levels of SA and, in turn, high levels of human performance; however, multimodal interaction within the VE must be appropriately designed to lead to this enhanced awareness. Specifically, the number of sensory modalities stimulated and the quality of this multisensory interaction are critical to the immersiveness and potential effectiveness of VE systems (Popescu, Burdea, and Trefftz 2002). There are some emerging guidelines that can guide the design of such multimodal interaction. For example, Stanney et al. (2004) provided a set of preliminary crossmodal integration rules. These rules consider aspects of multimodal interaction, including (1) temporal and spatial coincidence, (2) working memory capacity, (3) intersensory facilitation effects, (4) congruency, and (5) inverse effectiveness. When multimodal sensory information is provided to users, it is essential to consider such rules governing the integration of multiple sources of sensory feedback. VE users have adapted their perception–action systems to "expect" a particular type of information flow in the real world; VEs run the risk of breaking these perception–action couplings if the full range of sensory is not supported or if it is supported in a manner that is not contiguous with real-world expectations. Such pitfalls can be avoided through consideration of the coordination between sensing and user command and the transposition of senses in the feedback loop. Specifically, command coordination considers user input as primarily monomodal and feedback to the user as multimodal. Designers need to consider which input modalities are most appropriate to support execution of a given task within the VE, if there is any need for redundant user input, and whether or not users can effectively handle such parallel input (Stanney, Mourant, and Kennedy 1998; Stanney et al. 2004). Additional multimodal design guidelines have been provided by Hale et al. (2009), who have outlined how a number of sensory cues may effectively be used to enhance specific SA components (i.e., object recognition, spatial, temporal) within a VE, with the goal of optimizing SA development.

A limiting factor in supporting multimodal sensory stimulation in VEs is the current state of interface technologies. With the exception of the visual modality, current levels of technology simply cannot even begin to reproduce virtually those sensations, such as haptic and audition, that users expect in the real world. One solution to current technological shortcomings, sensorial transposition, occurs when a user receives feedback through senses other than those expected, which may occur because a command coordination scheme has substituted available sensory feedback for those that cannot be generated within a VE. Sensorial substitution schemes may be one for one (e.g., visual for force) or more complex (e.g., visual for force and auditory; visual and auditory for force). If designed effectively, command coordination and sensory substitution schemes should provide multimodal interaction that allows for better user control of the VE. However, if designed poorly, these solutions may in fact exacerbate interaction problems.

28.3.1.2 Perceptual Illusions

When sensorial transpositions are used, there is an opportunity for perceptual illusions to occur. With perceptual illusions, certain perceptual qualities perceived by one sensory system are influenced by another sensory system (e.g., "feel" a squeeze when you see your hand "grabbing" a virtual object). Such illusions could simplify and reduce the cost of VE development efforts (Storms 2002). For example, when attending to a visual image coupled with a low-quality auditory display, auditory–visual crossmodal perception allows for an increase in the perceived quality of the visual image. Thus, in this case if the visual image is the focus of the task, there may be no need to use a high-quality auditory display.

There are several types of perceptual illusions that can be used in the design of VEs (Steinicke and Willemsen 2010). Visual illusions can be used to substitute for missing proprioceptive and vestibular senses, as vision usually dominates these senses. For example, vection (i.e., a compelling illusion of self-motion throughout a virtual world) is known to be enhanced via a number of visual display factors, including a wide FOV and high spatial frequency content (Hettinger 2002), as well as visual jitter (Kitazaki, Onimaru, and Sato 2010). In addition, change blindness (i.e., failing to notice alterations in a visual scene) can be used to apply subtle manipulations to the geometry of a VE and direct movement behavior, such as redirecting a user's walking path throughout a VE (Suma et al. 2010). Other such illusions exist and could likewise be leveraged if perceptual and cognitive design principles are identified that can be used to trigger and capitalize on these illusory phenomena. For example, acoustic illusions (e.g., a fountain sound; Riecke, Väljamäe, and Schulte-Pelkum 2009) could also be used to create a sense of vection in a VE, even when no such visual motion is provided. In addition, haptic illusions (Hayward 2008) could be used to provide users with the impression of actually feeling virtual objects when they are in fact touching real-world props or traveling along a trajectory path that may even vary in size, shape, weight, or surface from their virtual counterparts without users perceiving these discrepancies (e.g., feel an illusory bump when actually touching a flat surface [Robles-De-La-Torre and Hayward 2001]; feel an illusory sharp edge when hand actually travels along a smooth trajectory [Portillo-Rodríguez et al. 2006]).

28.3.1.3 Navigation and Wayfinding

Effective multimodal interaction design and use of perceptual illusions can be impeded if navigational complexities arise. Navigation is the aggregate of wayfinding (e.g., cognitive planning of one's route) and the physical movement that allows travel throughout a VE (Darken and Peterson 2002). A number of tools and techniques have been developed to aid wayfinding in virtual worlds, including maps, landmarks, trails, and direction finding. These tools can be used to display current position, current orientation (e.g., compass), log movements (e.g., "breadcrumb" trails), demonstrate or access the surround (e.g., maps, binoculars), or provide guided movement (e.g., signs, landmarks) (Chen and Stanney 1999). For example, Burigat and Chittaro (2007) found 3D arrows to be particularly effective in guiding navigation throughout an abstract VE. Darken and Peterson (2002) provided a number of principles concerning how best to use these tools. If effectively applied to VEs, these principles should lead to reduced disorientation and enhanced wayfinding in large-scale VEs.

28.3.2 Content Development

Content development is concerned with the design and construction of the virtual objects and synthetic environment that support a VE experience (Isdale et al. 2002). Although this medium can leverage existing HCI design principles, it has unique design challenges that arise due to the demands of real-time, multimodal, collaborative interaction. In fact, content designers are just starting to appreciate and determine what it means to create a full-sensory experience with user control of both point of view and narrative development. Aesthetics is thought to be a product of agency (e.g., pleasure of being), narrative potential, presence and co-presence (e.g., existing in and sharing the virtual experience), and transformation (e.g., assuming another persona) (Murray [1997]). Content development should be about stimulating perceptions (e.g., sureties, surprises), and contemplation over the nature of being (Isdale et al. 2002).

Existing design techniques, for example, from entertainment, video games, and theme parks, can be used to support VE content development (see Chapters 31 and 34). Game development techniques that can be leveraged in VE content development include but are not limited to providing a clear sense of purpose, emotional objectives, perceptual realism, intuitive interfaces, multiple solution paths, challenges, a balance of anxiety and reward, and an almost unconscious flow of interaction (Isdale et al. 2002). From theme park design, content development suggestions include (1) having a story that provides the all-encompassing theme of the VE and thus the "rules" that guide the design, (2) providing location and purpose, (3) using cause-and-effect to lead users to their own conclusions, and (4) anchoring users in the familiar (Carson 2000a,b). Although these suggestions provide guidelines for VE content development, considerable creativity is still an essential component of the process. Isdale et al. (2002) suggested that the challenges of VE content development highlight the need for art to compliment technology.

> I expect that the user experience of VR in 2025 will be something close to seamless; that is, the person will experience little if anything that intrusively indicates that 'this experience is a simulation and not reality' (Koltko-Rivera 2005).

Although the content incorporated into the virtual worlds of today is mostly quite separate from the real world, in recent years, life and technology have been more tightly coupled, the result being that computers are starting to have an awareness of themselves and the people who interact with them in 3D virtual spaces are evolving into a "second life." Virtual worlds are in fact penetrating our native space, and content development for future generations will likely aim to allow us to seamlessly use our own native language, with its wide range of verbal and physical gestures and emotions, thereby more fully entwining our first and second (virtual) lives (Rolston 2010).

28.3.3 Products Liability

Those who implement VE systems must be cognizant of potential product liability concerns. Exposure to a VE system often produces unwanted side effects that could render users incapable of functioning effectively upon return to the real world. These adverse effects may include nausea and vomiting, postural instability, visual disturbances, and profound drowsiness (Stanney et al. 1998). As users subsequently take on their normal routines, unaware of these lingering effects, their safety and well being may be compromised. If a VE product occasions such problems, liability of VE developers or system administrators could range from simple accountability (e.g., reporting what happened) to full legal liability (e.g., paying compensation for damages) (Kennedy and Stanney 1996; Kennedy, Kennedy, and Bartlett 2002). To minimize their liability, manufacturers and corporate users should design systems and provide usage protocols to minimize risks, warn users about potential aftereffects, monitor users during exposure, assess users' risk, and debrief users after exposure.

28.3.4 Usage Protocols

To minimize product liability concerns, VE usage protocols should be carefully designed. A comprehensive VE usage protocol will involve the following activities (see Stanney, Graeber, and Kennedy [2005]):

1. Designing VE stimulus to minimize adverse effects:
   a. Ensure system lags/latencies are minimized and stable

   b. Ensure frame rates are optimized
   c. Ensure interpupilary distance (IPD) of visual display is adjustable and set appropriately
   d. Determine if size of FOV is causing excessive vection, such that the spatial frequency content of the visual scene should be reduced
   e. Determine if multimodal feedback is integrated into the VE such that sensory conflicts are minimized
2. Quantifying stimulus intensity of a VE system using the Simulator Sickness Questionnaire (Kennedy et al. 1993) or other means and comparing the outcome with other systems (see Stanney et al. [2005]). If a given VE system is of high intensity (say the fiftieth or higher percentile) and is not redesigned to lessen its impact, significant dropouts can be expected.
3. Identifying individual capacity of target user population to resist adverse effects of VE exposure via the Motion History Questionnaire (Kennedy and Graybiel 1965) or other means.
   a. For highly susceptible populations, redesigning the VE to reduce stimulus intensity or expect high dropout rates
   b. Providing warnings for those with severe susceptibility to motion sickness, seizures, migraines, cold, flu, or other ailments
4. Setting exposure duration and intersession interval to minimize adverse effects by (see Stanney et al. [2005]) the following:
   a. Limiting initial exposures (e.g., 10 minutes or less)
   b. Setting intersession exposure intervals 2–5 days apart to enhance individual adaptability
   c. Incrementally increasing VE stimulus intensity within one exposure (incremental adaptation) or across multiple exposures (incremental habituation)
   d. Determining if users are able to complete an adaptive process at each increment of stimulus intensity (e.g., depressing sensitization to pre-exposure levels or below); if not, lowering stimulus intensity
   e. Avoiding repeated exposure intervals occurring less than 2 hours apart if adverse effects are experienced in an exposure
5. Educating users regarding potential risks of VE exposure (e.g., inform users they may experience nausea, malaise, disorientation, headache, dizziness, vertigo, eyestrain, drowsiness, fatigue, pallor, sweating, increased salivation, and vomiting).
6. Educating users regarding potential adverse aftereffects of VE exposure (e.g., inform users they may experience disturbed visual functioning, visual flashbacks, and unstable-locomotor and postural control for prolonged periods post exposure).
7. Instructing users to terminate VE interaction if they start to feel ill.
8. Providing adequate air flow and comfortable thermal conditions.
9. Adjusting equipment to minimize fatigue.
10. For strong VE stimuli, warning users to avoid extraordinary maneuvers (e.g., flying backward or experiencing high rates of linear or rotational acceleration) during initial interaction.
11. Providing an attendant to monitor users' behavior and ensure their well being:
    a. Attendant should terminate exposure if they see red flags (e.g., excessive sweating, verbal frustration, prolonged lack of movement in VE, and less overall movement [e.g., restricting head movement] or if users verbally request termination or complain of excessive symptoms).
    b. Attendant should assess well being of users' postexposure (e.g., can use battery similar to field sobriety tests; Tharp, Burns, and Moskowitz [1981]).
12. Specifying amount of time postexposure that users must remain on premises before driving or participating in other such high-risk activities. Do not allow individuals who fail postexposure tests or experience adverse aftereffects to conduct high-risk activities until they have recovered (e.g., have someone else drive them home).
13. Calling users the next day or having them call to report any prolonged adverse effects.

Regardless of the strength of the stimulus or the susceptibility of the user, following a systematic usage protocol can minimize the adverse effects associated with VE exposure.

28.4 HEALTH AND SAFETY ISSUES

The health and safety risks associated with VE exposure complicate usage protocols and lead to product liability concerns. It is thus essential to understand these issues when utilizing VE technology. There are both physiological and psychological risks associated with VE exposure, the former being related primarily to sickness and aftereffects and the latter primarily being concerned with the social impact.

28.4.1 Cybersickness, Adaptation, and Aftereffects

Motion-sickness-like symptoms and other aftereffects (e.g., balance disturbances, visual stress, altered hand–eye coordination) are unwanted byproducts of VE exposure (Stanney and Kennedy 2008). The sickness related to VE systems is commonly referred to as "cybersickness" (McCauley and Sharkey 1992). Some of the most common symptoms exhibited include dizziness, drowsiness, headache, nausea, fatigue, and general malaise (Kennedy et al. 1993). More than 80% of users will experience some level of disturbance, with approximately 12% ceasing exposure prematurely due to this adversity (Stanney et al. 2003). Of those who drop out, approximately 10% can be expected to have an emetic response (e.g., vomit); however, only 1–2% of all users will have such a response. These adverse effects are known to increase in incidence and intensity with prolonged exposure duration (Kennedy, Stanney, and Dunlap 2000). Although most users will experience some level of adverse effects, symptoms vary substantially from one individual to another and from one system to another (Kennedy and Fowlkes 1992). These effects can be assessed via the Simulator Sickness Questionnaire (Kennedy et al. 1993), with values above 20 requiring due caution (e.g., warn and observe users) (Stanney et al. [2005]).

To overcome such adverse effects, individuals generally undergo physiological adaptation during VE exposure. This adaptation is the natural and automatic response to an intersensorily imperfect VE and is elicited due to the plasticity of the human nervous system (Welch 1978). Because of technological flaws (e.g., slow update rate, sluggish trackers), users of VE systems may be confronted with one or more intersensory discordances (e.g., visual lag, a disparity between seen and felt limb position). To perform effectively in the VE, they must compensate for these discordances by adapting their psychomotor behavior or visual functioning. Once interaction with a VE is discontinued, these compensations persist for some time after exposure, leading to aftereffects.

Once VE exposure ceases and users return to their natural environment, they are likely unaware that interaction with the VE has potentially changed their ability to effectively interact with their normal physical environment (Stanney and Kennedy 1998). Several different kinds of aftereffects may persist for prolonged periods following VE exposure (Welch 1997). For example, hand–eye coordination can be degraded via perceptual-motor disturbances (Kennedy et al. 1997; Rolland et al. 1995), postural sway can arise (Kennedy and Stanney 1996), as can changes in the vestibulo-ocular reflex, or one's ability to stabilize an image on the retina (Draper, Prothero, and Viirre 1997). The implications of these aftereffects are the following:

1. VE exposure duration may need to be minimized.
2. Highly susceptible individuals or those from clinical populations (e.g., those prone to seizures) may need to avoid or be banned from exposure.
3. Users should be closely monitored during VE exposure.
4. Users' activities should be closely monitored for a considerable period of time postexposure to avoid personal injury or harm.

28.4.2 Social Impact

VE technology, like its ancestors (e.g., television, computers), has the potential for negative social implications through misuse and abuse Kallman [1993]. Yet violence in VE is nearly inevitable, as evidenced by the violent content of popular video games. Such animated violence is a known favorite over the portrayal of more benign emotions such as cooperation, friendship, or love (Sheridan [1993]; also see Chapter 5. The concern is that users who engage in what seems like harmless violence in the virtual world may become desensitized to violence and mimic this behavior in the look-alike real world.

Currently, it is not clear whether or not such violent behavior will result from VE exposure; early research, however, is not reassuring. Calvert and Tan (1994) found VE exposure to significantly increase the physiological arousal and aggressive thoughts of young adults (Calvert and Tan 1994). Perhaps, more disconcerting was that neither aggressive thoughts nor hostile feelings were found to decrease due to VE exposure, thus providing no support for catharsis. Such increased negative stimulation may then subsequently be channeled into real-world activities. The ultimate concern is that VE immersion may potentially be a more powerful perceptual experience than past, less interactive technologies, thereby increasing the negative social impact of this technology (Calvert 2002). A proactive approach is needed, which weighs the risks and potential consequences associated with VE exposure against the benefits. Waiting for the onset of harmful social consequences should not be tolerated. Koltko-Rivera (2005) suggests that a proactive approach would involve determining the (1) types and degree of VE content (e.g., aggressive, sexual, etc.), (2) types of individuals or groups exposed to this content (e.g., their mental aptitude, mental conditioning, personality, world-view, etc.), (3) circumstances of exposure (e.g., private experience, family, religion, spiritual, etc.), and (4) effects of exposure on psychological, interpersonal, or social function.

28.5 VIRTUAL ENVIRONMENT USABILITY ENGINEERING

Most VE user interfaces are fundamentally different from traditional graphical user interfaces, with unique I/O devices, perspectives, and physiological interactions. Thus, when developers and usability practitioners attempt to apply traditional usability engineering methods to the evaluation of VE systems, they find few if any that are particularly well suited to these environments (for notable exceptions, see Gabbard, Hix, and Swan [1999]; Hix and Gabbard [2002]; Stanney, Mollaghasemi, and Reeves [2000]). There is a need to modify and optimize available techniques to meet the needs of VE usability engineering and to better characterize factors unique to VE usability, including sense of presence and VE ergonomics.

28.5.1 Usability Techniques

Assessment of usability for VE systems must go beyond traditional approaches, which are concerned with the determination of effectiveness, efficiency, and user satisfaction (Bowman, Gabbard, and Hix [2002]; see Chapter 53). Evaluators must consider whether multimodal input and output is optimally presented and integrated, navigation is supported to allow the VE to be readily traversed, object manipulation is intuitive and simple, content is immersive and engaging, and the system design optimizes comfort while minimizing sickness and aftereffects. The affective elements of interaction become important when evaluating VE systems (see Chapter 31). It is an impressive task to ensure that all these criteria are met.

Gabbard, Hix, and Swan (1999) have developed a taxonomy of VE usability characteristics that can serve as a foundation for identifying and evaluating usability criteria particularly relevant to VE systems. Stanney, Mollaghasemi, and Reeves (2000) used this taxonomy as the foundation on which to develop an automated system, Multicriteria Assessment of Usability for VE (MAUVE), which assesses VE usability in terms of how effectively each of the following are designed: (1) navigation, (2) user movement, (3) object selection and manipulation, (4) visual output, (5) auditory output, (6) haptic output, (7) presence, (8) immersion, (9) comfort, (10) sickness, and (11) aftereffects. MAUVE can be used to support expert evaluations of VE systems, similar to the manner in which traditional heuristic evaluations are conducted. Because of such issues as cybersickness and aftereffects, it is essential to use these or other techniques (cf., Modified Concept Book Usability Evaluation Methodology; Swartz [2003]) to ensure the usability of VE systems, not only to avoid rendering them ineffective but also to ensure that they are not hazardous to users. Recently, guidelines have been evolving for enhancing the design of social VEs (e.g., Second Life by Linden Labs, Whyville by Numedeon, Inc.), such as those promoted by the Center for Disease Control and Prevention for reaching individuals with timely health information that may relate to campaigns and upcoming events (CDC 2010).

28.5.2 Sense of Presence

One of the usability criteria unique to VE systems is sense of presence. VEs have the unique advantage of leveraging the imaginative ability of individuals to psychologically “transport” themselves to another place, one that may not exist in reality (Sadowski and Stanney 2002). To support such transportation, VEs provide physical separation from the real world by immersing users in the virtual world via an HMD, then imparting sensorial sensations via multimodal feedback that would naturally be present in the alternate environment. Focus on generating such presence is one of the primary characteristics distinguishing VEs from other means of displaying information.

Presence has been defined as the subjective perception of being immersed in and surrounded by a virtual world rather than the physical world one is currently situated in (Stanney et al. 1998). VEs that engender a high degree of presence are thought to be more enjoyable, effective, and well received by users (Sadowski and Stanney 2002). High presence VEs are also suggested to be effective learning environments (Mantovani and Castelnuovo 2003), as well as to enhance behavioral modeling outcomes and lead to greater imitation in the physical world (Fox, Bailenson, and Binney 2009). To enhance presence, designers of VE systems should spread detail around a scene, let user interaction determine when to reveal important aspects, maintain a natural and realistic, yet simple appearance, and utilize textures, colors, shapes, sounds, and other features to enhance realism (Kaur 1999). To generate the feeling of immersion within the environment, designers should isolate users from the physical environment (use of an HMD may be sufficient), provide content that involves users in an enticing situation supported by an encompassing stimulus stream, provide natural modes of interaction and movement control, and utilize design features that enhance vection (Stanney, Mollaghasemi, and Reeves 2000). To enhance presence in learning environments, the design of perceptual features (i.e., perceptual realism, interactivity, and control), individual factors (i.e., imagination and suspension of disbelief, identification, motivation and goals, and emotional state), content characteristics (i.e., plot, story, narration, and dramaturgy), and interpersonal, social, and cultural context should be carefully considered (Mantovani and Castelnuovo 2003). Presence can be assessed via Witmer and Singer’s (1998) Presence Questionnaire or techniques used by Slater and Steed (2000), as well as by a number of other means (Sadowski and Stanney 2002).

28.5.3 Virtual Environment Ergonomics

Ergonomics, which focuses on fitting a product or system to the anthropometric, musculoskeletal, cardiovascular, and psychomotor properties of users, is an essential element of VE system design (McCauley-Bell 2002). Supporting user comfort while donning cumbersome HMDs or unwieldy peripheral devices is an ergonomics challenge of paramount importance; discomfort could supersede any other sensations (e.g., presence, immersion). If a VE produces discomfort, participants may limit their exposure time or possibly avoid repeat exposure. Overall, physical discomfort should thus be minimized, while user safety is maximized.

Ergonomic concerns affecting comfort include visual discomfort resulting from visual displays with improper depth cues, poor contrast and illumination, or improperly set Ensure interpupilary distance IPDs (Stanney et al. 2000). Physical discomfort can be driven by restrictive tethers; awkward interaction devices; or heavy, awkward, and constraining visual displays. To enhance the ergonomics of VE systems, several factors should be considered, including (McCauley-Bell 2002) the following:

- Is operator movement inhibited by the location, weight, or window of reach of interaction devices or HMDs?
- Does layout and arrangement of interaction devices and HMDs support efficient and comfortable movement?
- Is any limb overburdened by heavy interaction devices or HMDs?

- Do interaction devices require awkward and prolonged postures?
- If a seat is provided, does it support user movement and is it of the right height with adequate back support?
- If active motion interfaces are provided (e.g., treadmills), are they adjustable to ensure fit to the anthropometrics of users?
- Are the noise and sound levels within ergonomic guidelines and do they support user immersion?

28.6 APPLICATION DOMAINS

The wide range of VE types and designs makes them ideal tools for supporting a range of performance-enhancing tasks. Most commonly, VEs are used as training aids; a less common but increasingly well-supported use is as a selection tool, the flip side to the training application. Other applications for VE include their growing potential for providing entertainment, their use as medical rehabilitative tools, and even their implementation within the training system acquisition cycle of the U.S. government.

28.6.1 Virtual Environment as a Selection and Training Tool

Perhaps the most common application for VEs is as training tools. The U.S. military has focused considerable resources on VE as a means to resolve many of the training deficits that result from the rigors of military life (e.g., sustaining schoolhouse-gleaned skills and knowledge sets during prolonged deployment periods, acquiring new skills and knowledge while away from the schoolhouse, providing large-scale environments for multitrainee distributed training exercises. Military VE training applications have been developed in a wide variety of areas, such as perceptual and cognitive performance (Carroll et al. 2010), decision making under stress (Carroll et al. 2010; Hill et al. 2003), operational readiness (Barba et al. 2006), and cross-cultural communication (Deaton et al. 2005; Stanney et al. 2010). Similarly, VE applications are being used as interactive tools for teaching medical students, nurses, and doctors such knowledge and skills as varied as the basics of human anatomy, complicated surgical procedures, communication skills, and decision-making skills (Grantcharov et al. 2004; Johnsen et al. 2005; Segal and Fernandez 2009; Fried et al. 2010; Hassinger et al. 2010).

As VE training exercises become more closely aligned to real-world conditions, a series of factors, including training transfer effectiveness, cost, logistics, and safety, move to the forefront. Further, in terms of applying VE to enhance human performance, training is actually the second stage of a two-stage process. Ideally, one would like to ensure that those individuals for whom training is being provided have a certain degree of "performance capability." In this approach, VE systems can be used as part of a comprehensive performance-enhancement program that focuses on selecting those users with the correct set of knowledge, skills, and abilities (KSAs) and then providing, when needed, training to fine-tune those KSAs.

Given the potentially high cost of developing VEs, this makes sense. Given the broader, yet equally high costs of bringing individuals into an organization and training them to some level of proficiency, it becomes glaringly obvious that having an ability to select individuals who are most likely to benefit from VE training is critical. For example, in the U.S. Air Force, the cost of a single individual student pilot failing to complete basic flight school can run to $100,000 (Siem, Carretta, and Mercatante 1988). Student failure can be attributed to both inadequate selection techniques and deficient training techniques. Clearly, both selection and training play critical roles in producing effective users. The challenge is to develop a program that ensures a smooth union between the two, which identifies the best candidates and then provides the optimum training.

Traditional approaches to selection focus on social and psychophysical assessments. For example, aptitude tests ranging from traditional pen-and-paper-type efforts (Carretta and Ree 1995) to psychomotor tests (Carretta and Ree 1993) to computer-based (but not VE) assessments (Carretta and Ree 1993) have all been used with varying levels of success. The single most important criticism of each of these approaches is that they are designed to be predictive of future performance and as such are more often than not abstractions of aspects of the larger task(s) for which the individual is being selected. An alternate approach would be to provide selectees with a method that provides a direct indication of their performance abilities. This distinction, essentially between a test being predictive of performance ability versus indicative of performance ability, has a great impact on selection. A meta-analysis performed within the aviation domain, where much of selection research has focused, found that typical predictive validities (most often reported as either the correlation coefficient, r, or the multiple correlation coefficient, R, and representing the degree to which a given predictor/set of predictors and performance metrics are related) for such assessments range from a low of 0.14 to a "high" of about 0.40 (Martinussen 1996). Yet, when a simulation component is added to this mix, these values have been shown to improve considerably, pushing correlations toward the 0.60 level (Gress and Willkomm 1996).

A potential deterrent with using VEs as part of a selection toolkit is the high cost associated with developing both the system and the performance measures that need to be integrated with it. However, when considered as one part of a comprehensive performance-enhancement program, this concern vanishes. The same effort and cost devoted to developing the system for selection can typically, with minimal additional input, be modified (e.g., via altered scenarios) to meet identified training goals and objectives. It is, therefore, necessary to explore some of the basic requirements for using VEs as training tools to understand how one technology can truly be applied for these multiple purposes.

A constant thread in training research is the notion that in order for training to be effective, the basic skills being taught must show some degree of transfer to real-world performance.

Over 100 years ago, Thorndike and Woodworth (1901) laid down the most basic training–transfer principle when they proposed that transfer was determined by the degree of similarity between any two tasks. Applying this heuristic to VE design, one might conclude that the most basic way to ensure perfect transfer is to ensure that the real-world performance elements that are meant to be trained should be replicated perfectly in a VE. This notion of "identical elements" could easily create a serious challenge for system designers even by today's technology standards, as VEs are still not able to perfectly duplicate the wide range of sensorial stimuli encountered during daily interactions with our world (Stoffregen et al. 2003). Countering this somewhat simplistic design approach is Osgood's (1949) principle that greater similarity between any two tasks along certain dimensions will not guarantee wholesale, perfect transfer. The challenge, as noted by Roscoe (1982), is to find the right balance between technical fidelity and training effectiveness.

This suggests that when developing training systems, it is important to take two factors into consideration. The first is how one can measure training transfer, and the second is how one can assess the cost/benefit relationship associated with making the decision to use VEs to provide training. The most basic metric for assessing transfer is the percent of transfer (Wickens and Hollands 2000) in which the amount of time saved in learning the real-world task is expressed in terms of how much time a control group, denied such training, required to learn the task (Equation 28.1):

$$\%\text{Transfer} = \left\{\frac{\text{Control time - Transfer time}}{\text{Control time}}\right\} \times 100 \quad (28.1)$$

A quick inspection of this metric suggests that it is overly simplistic. Consider a situation in which the percent transfer is high, yet the total amount of time spent training in the transfer platform (e.g., VE) may be greater than that spent by a control group to achieve similar levels of performance. Such an outcome is hardly cost-effective. Consequently, in parallel to a basic quantification of transfer, another metric, the training effectiveness ratio (TER) (Povenmire and Roscoe [1973]) focusing on training efficiency is needed (Equation 28.2).

$$\frac{\text{TER} = \text{Amount of time saved learning task by transfer group}}{\text{Time spent in transfer platform}} \quad (28.2)$$

All things being equal, these two measures should perfectly quantify VE utility as a training tool. Yet, a third factor, cost, almost always factors in. Indeed, cost is often the single most critical driving factor in determining whether to supplement a training program with VE tools. In terms of training effectiveness, cost can be quantified using the training cost ratio (TCR) (Wickens and Hollands [2000]), as shown in Equation 28.3.

$$\frac{\text{TCR} = \text{Training cost in real platform}}{\text{Training cost in transfer platform}} \quad (28.3)$$

When making the determination to use a VE to provide some (or, less likely, all) of a given set of training, one must consider the interplay between these measures. Specifically, the relationship between training effectiveness, as expressed through the TER, and training cost, as expressed through the TCR, will drive this decision. A good rule of thumb, provided by Wickens and Holland (2000), is

$$\text{TER} \times \text{TCR} > 1.0, \text{VE Training is effective} \quad (28.4)$$

$$\text{TER} \times \text{TCR} < 1.0, \text{VE Training is ineffective} \quad (28.5)$$

A slightly different way of interpreting this was provided by Roscoe (1982), who suggested that, given that the higher the level of technical fidelity the more costly the system, there is a region of design space within which cost/benefit is optimal; move away from this region in either direction and you run the risk of paying a lot for a system that delivers less than effective (and perhaps even negative) training.

This brief overview should not leave one with the impression that the decision to use—or to not use—VEs for training is simple. Assessing the TER × TCR function is no easy matter. Simply put, the cost of performing the study needed to capture the effectiveness of a training device can far outstrip the perceived immediate impact of incorporating the device into a training curriculum. Often, the type of system being developed requires a significant investment in time, money, and labor to be evaluated. Consider a situation in which a new flight simulator is being considered for purchase. To determine both the TER and the TCR, one must conduct an actual transfer study, which involves the following:

1. Removing a set of trainees from their curriculum for experimental purposes
2. Developing reliable performance metrics
3. Providing both the experiment group and the control group with adequate time in the real-world (e.g., aircraft) environment and the experiment group time in the VE
4. Dealing with typically low-effect sizes, which forces experimenters to use large sample sizes

Yet, just as the TER × TCR function could guide when to use and not use VEs, it is possible to pull some general principles from the literature. For example, cognitively oriented training, which often features problem solving or decision making as a key training goal, may oftentimes be satisfied by using simple, relatively inexpensive visualization systems (Gopher, Weil, and Bareket 1994; Morris and Tarr 2002; Stone 2002; Figueroa, Bischof, Boulanger, and Hoover 2005; Milham et al. 2004). Contrastingly, motor skill–based training, which often requires complex interactions between users and their environments, frequently requires more costly solutions. When considering whether or not to "go virtual," a useful solution would be to first determine the general category of skill to be trained (Cognitive or Motor; cf., Anderson

[1987]); estimate the TER and TCR from sources as broadly ranging as basic science studies, published costs of similarly desired systems, and one's own best estimates; and then develop a basic decision matrix to establish a range of TER × TCR values.

28.6.2 Virtual Environment as an Entertainment and Education Tool

VEs have reached beyond their original applications, primarily as military training tools, and have extended into a wide variety of entertainment applications. From interactive arcades to cybercafes, the entertainment industry has leveraged the unique characteristics of the VE medium, providing dynamic and exciting experiences in a multitude of forms. VE entertainment applications have found their way into games, sports, movies, art, online communities, location-based entertainment, theme parks, and other venues (Badiqué et al. 2002; Nakatsu, Rauterberg, and Vorderer 2005). By exploiting the unique interactive characteristics of VEs compared with more traditional entertainment media (e.g., film, play), VE technology provides a more immersive medium for entertainment through the use of simple artificial virtual characters (i.e., avatars), engaging narrative and dynamic control to create an immersive interactive experience.

Interestingly, the utility of virtual games has been found to have a cultural differential. Rauterberg (2009) found that while in Western countries game content is focused on violence, in Asia, they are exploring intensive game usage and its impact on the intellectual development of children. Yet, this is likely changing, as evidenced by the recent focus on educational games in the West (cf., the eEducation Roadmap developed by the National Aeronautics and Space Administration [NASA] eEducation and the Federation of American Scientists, which explored the research challenges associated with the design of immersive environments for education; Laughlin, Roper, and Howell [2007]). This report suggests that it is essential to understand how best to exploit the unique features of synthetic gaming environments by characterizing which features of these systems are important for learning and why. Specific research challenges that the report indicates must be met to realize the full potential of immersive game-based learning environments that enhance education including the following:

- Developing an understanding of effective learning strategies based on the needs of individual learners and how to use this information to design an individualized, immersive learning experience
- Determining how best to take advantage of the benefits offered by immersive gaming technologies to facilitate educational inquiry (i.e., enhancing the frequency and quality of questions posed by learners) and effective methods for delivering answers to the questions posed
- Determining how best to deliver feedback and guidance and use real-time assessment to adapt the immersive learning environment to the needs of individual learners such that they proceed to mastery in the most effective and efficient manner
- Determining how to design effective collaborative VEs that reflect current understanding of physics, chemistry, biology, mathematics, and other disciplines that permit collaborative exploration-based pedagogy

Until recently, the high cost of the component technologies comprising VEs precluded their becoming anything more than a research test-bed for well-funded laboratories or an exclusive tool for high-end consumers, like commercial airline companies or the military establishments of entire countries. Yet, after nearly 5 decades of research and development, VEs have finally begun to realize their potential and are making their way into both entertainment and educational applications. It is likely only the imagination that limits future such applications of VE technology.

28.6.3 Virtual Environment as a Medical Tool

> What makes VR application development in the assessment, therapy, and rehabilitation sciences so distinctively important is that it represents more than a simple linear extension of existing computer technology for human use. VR offers the potential to create systematic human testing, training, and treatment environments that allow for the precise control of complex, immersive, dynamic three-dimensional (3-D) stimulus presentations, within which sophisticated interaction, behavioral tracking and performance recording is possible. (Rizzo, et al. 2006, p. 36)

Much has been written about applications for VE within the medical arena (cf., Moline 1995; Satava and Jones [2002]). Although some of these applications represent unique approaches to harnessing the power of VE, many other applications, such as simulating actual medical procedures, reflect training applications, and therefore will not be discussed anew here. One area of medical application for which VE is truly coming into its own is medical rehabilitation. In particular, two areas of rehabilitation, behavioral/cognitive and motor, show strong promise.

28.6.3.1 Behavioral/Cognitive Rehabilitation Applications

In terms of behavioral rehabilitation applications, VE applications have been gaining prominence in behavioral science research over the past several years. For example, VE cue reactivity programs have been successfully tested for feasibility in nicotine-(Bordnick et al. 2005), opiate-(Kuntze et al. 2001), and alcohol-(Bordnick et al. 2008) dependent individuals, and those with eating disorders (Gutiérrez-Maldonado, Ferrer-García, Caqueo-Urizar, and Letosa-Porta 2006). VE applications have also shown promise in modifying exercise behavior (Fox and Bailenson 2009), retirement savings behavior (Ersner-Hershfield, Bailenson, and Carstensen 2008), and managing pain (Dahlquist et al. 2010; Gold, Belmont, and Thomas 2007; Hoffman et al. 2008). Perhaps the fastest-growing application for VEs in behavioral rehabilitation is in the area of exposure therapy (Fox, Arena, and Bailenson 2009; Gregg and Tarrier 2007; Parsons and Rizzo 2008; Powers and Emmelkamp 2008). For example, VE applications have been used to treat acrophobia (the fear of heights; Coelho et al. 2006), agoraphobia (fear of open spaces; Botella et al. 2007), arachnophobia (fear of spiders; Cote and Bouchard 2005); aviophobia (fear of flying; Rothbaum et al. 2000), combat-related posttraumatic stress disorder (Reger and Gahm 2008), panic disorder (Botella et al. 2007), public speaking anxiety (Harris, Kemmerling, and North 2002), and social phobia (Roy et al. 2003). The reason for this broad use of VE technology for exposure therapy is likely due to the ideal matching between VE's strengths (presenting evolving information with which users can interact in various ways) and such therapy's basic requirements (incremental exposure to the offending environment). Importantly, compared with previous treatment regimens, which oftentimes simply required patients to mentally revisit their fears, VEs offer a significantly more immersive experience. In fact, it is quite likely that many of VE's shortcomings, such as poor visual resolution, inadequate physics modeling underlying environmental cues, and failure to fully capture the wide range of sensorial cues present in the real world will be ignored by the patient, whose primary focus is on overcoming anxiety engendered by her or his specific phobias. On a practical level, VEs enable patients to simply visit their therapist's office, where they can be provided an individually tailored multimodal treatment experience (Rothbaum et al. 1996; Emmelkamp et al. 2001; Anderson, Rothbaum, and Hodges 2003).

Beyond behavioral rehabilitation, VE applications are being developed for the study, assessment, and rehabilitation of various types of cognitive processes, such as perception, attention, and memory. For example, VE applications are being used as perceptual skill trainers, such as for elderly drivers who have degraded visual scanning behavior (Romoser and Fisher 2009) and for rehabilitating stroke victims who suffer from Unilateral Spatial Neglect, where an individual fails to perceive stimuli presented to the contralesional hemi-visual field even though they are not "blind" to this area (Katz et al. 2005). In terms of attention, attention deficit hyperactivity disorder is an example of a cognitive dysfunction that has been addressed via VE rehabilitation applications (Parsons et al. 2007). Brooks and Rose (2003) suggest that VE rehabilitation applications can be used both in terms of assessment of memory impairments and memory remediation (e.g., use of reorganization techniques), where it has been found to promote procedural learning of those with memory impairments; importantly, this learning has been found to transfer to improved real-world performance. Examples of memory remediation in VEs include its use to enhance the ability of stroke victims to remember to perform actions in the future (Brooks et al. 2004) and its use in enhancing the performance of an individual with age-related impairment in memory-related cognitive processes (Optale et al. 2001). VEs have also been shown to uncover subtle cognitive impairments that might otherwise go undetected (Tippett et al. 2009). In general, VE applications can provide precisely controlled means of assessing cognitive impairments that are not available using more traditional evaluation methods. Specifically, VEs can deliver an assessment environment, where controlled stimuli can be presented at varying degrees of perception/attention/memory challenge, and level of deficit can be assessed. This level of experimental control allows for the development of both cognitive impairment assessment and rehabilitation applications that have a high level of specificity and ecological validity.

28.6.3.2 Motor Rehabilitation Applications

Many of VE's qualities that make it an ideal tool for providing medical training, such as tactile feedback and detailed visual information (Satava and Jones 2002), also make it an ideal candidate for supplementing motor rehabilitation treatment regiments for such conditions as stroke (Deutsch and Mirelman 2007; Yeh et al. 2007), cerebral palsy (Bryanton et al. 2006), and amblyopia (i.e., lazy-eye) (Eastgate et al. [2006]). Specifically, Fox, Arena, and Bailenson (2009) suggest that VEs have three features that make them uniquely suited to facilitating motor rehabilitation, including the ability to review one's physical behavior and interactively exam one's progress, see one's own avatar from a third-person perspective in real time, and safely recreate real environments that cannot otherwise be experienced (e.g., crossing a busy intersection). In determining how best to apply VE in physical rehabilitation treatment regimens, Holden (2005) suggested considering three practical areas in which VE is strongest: repetition, feedback, and motivation. All three elements are critical to both effective learning and regaining of motor function. The application of VE, in each case, provides a powerful method for rehabilitation specialists to maximize the effect of a treatment regimen for a given session and, because they may reduce the time investment required by therapists (one can simply immerse the patient, initiate the treatment, and then allow the program to execute), to also expand the access of such treatments to a wider population.

Since VE is essentially computer-based, patients can effectively have their attention drawn to a specific set of movement patterns they may need to make to regain function; conducting this in a "loop" provides unlimited ability to repeat a pattern while using additional visualization aids, such as a rendered cursor or "follow-me" types of cues, to force the patient into moving a particular way (cf., Chua et al. [2003]). As well, it is a relatively simple matter to digitize movement information, store it, and then, based on comparisons to previously stored, desired, movement patterns, provide additional feedback to assist the patient. In terms of motivation, treatment scenarios can be tailored to capture specific events that individual patients find most motivating: a baseball fan can practice her movement in a baseball-like scenario; a car enthusiast can do so in a driving environment.

There are certain caveats that must be considered when exploiting VE for rehabilitation purposes, most significantly the potentially rapid loss of motor adaptations following VE exposure. Lackner and DiZio (1994) demonstrated that certain basic patterns of sensori-motor recalibrations learned in a given physical environment can diminish within an hour, post-exposure, although subsequent findings (DiZio and Lackner 1995) suggest that there are certain transfer benefits that are longer-lasting. Brashers-Krugg, Shadmehr, and Bizzi (1996) provided additional evidence that sensori-motor recalibrations of the type likely to be required for rehabilitation have postexposure periods, in excess of 4 hours, during which their effects can be extinguished. Most concerning Cohn, DiZio, and Lackner (2000) demonstrated that such recalibrations, when learned in VE, have essentially no transfer to real-world conditions post exposure. Clearly, more research is needed to understand the conditions under which such transfer effects can be made most effective within the clinical setting.

28.6.4 Virtual Environment as a System Acquisition Tool

In many areas of research and development, rising costs, with their associated increasing consequences of failure, have all but forced scientists and engineers to focus only on developing those products that have an almost guaranteed chance of succeeding in the marketplace. Although this may seem like an ideal situation—after all, who would want to field technologies that won't work?—the net result is an increasing aversion, among companies and their financial backers, to shoulder much risk. Consequently, much-needed breakthrough technologies may be ignored in favor of "sure-thing" technologies, which more often than not do little to push ahead technology barriers.

The United States Department of Defense (DoD), seeking to reverse this trend while maintaining a level of control over risk, developed the process of simulation-based acquisition , a methodology that introduced the use of modeling and simulation (M&S) within the product-development process (Sanders 1997; Zittel 2001). Most recently, these M&S solutions have become increasingly integrated into 4D interactive tools that enable human-in-the-loop testing of design and operational principles (Sanders 1997). As one example that cuts across DoD and industry, consider the development of the Joint Strike Fighter (JSF). While more common M&S solutions, such as the ability to simulate individual components digitally before manufacturing them (through the CAD/CAM process), formed a core component of the development process, more advanced M&S solutions, such as VEs, have enabled developers to include users throughout the development process to answer such questions as (1) how users (e.g., pilots) would implement the fighter in combat, using virtual war-gaming techniques; (2) how well individual pilots could handle a given JSF design; and (3) how the JSF would alter the nature of warfare (Zittel 2001).

This is just a small glimpse of the potential applications for VE technology, the limit of which is bound only by our imagination. Recently, VEs have even been suggested as having potential value in simulating reenactments of criminal situations to enhance courtroom practices (Leonetti and Bailenson 2010). It will be interesting to see the vast variety of future VE applications that evolve.

28.7 CONCLUSIONS

Clearly, VE technology has evolved significantly over the past 5 decades. Yet, despite significant revolutions in component technology, many of the challenges addressed over this time period, such as multimodal sensori-interaction, scenario generation, and sickness and aftereffects, have yet to be fully resolved. At the same time, our understanding of the potential such tools have to offer has advanced considerably. No longer for simple amusement, this powerful technology can provide educational value, assist in treating core physical and cognitive maladies and even help design better interactive systems. As the uses for which VEs are ideally suited continue to be defined and refined, one can anticipate that current development challenges will be resolved, only to find new ones waiting to take their place.

ACKNOWLEDGMENTS

The authors thank Branka Wedell for her contribution of Figures 28.1 and 28.3 and Hesham Fouad for his contribution to the audio technology review within the Hardware Requirements section and the contribution of Figure 28.2. This material is based on work supported in part by the National Science Foundation (NSF) under Grant No. IRI-9624968, the Office of Naval Research (ONR) under Grant No. N00014-98-1-0642, the Naval Air Warfare Center Training Systems Division (NAWC TSD) under contract No. N61339-99-C-0098, and the National Aeronautics and Space Administration (NASA) under Grant No. NAS9-19453. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views or the endorsement of the NSF, ONR, NAWC TSD, or NASA.

REFERENCES

Acosta, E., and A. Liu. 2007. Real-time volumetric haptic and visual burrhole simulation. In *Proceedings of IEEE Virtual Reality Conference, VR 2007*, 247–50, Charlotte, NC. Los Alamitos, CA: IEEE Press.

Akamatsu, M. 1994. Touch with a mouse. A mouse type interface device with tactile and force display. In *Proceedings of 3rd IEEE International Workshop on Robot and Human Communication*, 140–4. Held July 18–20, Nagoya, Japan. Los Alamitos, CA: IEEE Press.

Algazi, V. R., R. O. Duda, D. M. Thompson, and C. Avendano. 2001. The CIPIC HRTF Database. In *Proceedings of the 2001 IEEE Workshop on Applications of Signal Processing to Audio and Electro-Acoustics*, 99–102, New Paltz, NY. Los Alamitos, CA: IEEE Press.

Allbeck, J. M., and N. I. Badler. 2002. Embodied autonomous agents. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 313–32. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Johnston, M. R., K. Hale, and P. Axelsson. 2010. Results from empirical testing of the System for Tactile Reception of Advanced Patterns (STRAP). In *Proceedings of 54th Human Factors and Ergonomics Society Annual Meeting*, September 27–October 1, San Francisco, CA. Thousand Oaks, CA: Sage Publications.

Anderson, J. R. 1987. Skill acquisition: Compilation of weak-method problem solutions. *Psychol Rev* 94:192–210.

Anderson, P., B. O. Rothbaum, and L. F. NS Hodges. 2003. Virtual reality exposure in the treatment of social anxiety. *Cognitive and Behavioral Practice* 10(3):240–7.

Badiqué, E., M. Cavazza, G. Klinker, G. Mair, T. Sweeney, D. Thalmann, and N. N. Thalmann. 2002. Entertainment applications of virtual environments. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 1143–66. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Ballas, J. A., D. Brock, J. Stroup, and H. Fouad. 2001. The effect of auditory rendering on perceived movement: Loudspeaker density and HRTF. In *Proceedings of the 2001 International Conference on Auditory Display*, 235–8. Held Jul 29–Aug 1, Espoo, Finland: Laboratory of Acoustics and Audio Signal Processing and the Telecommunications Software and Multimedia Laboratory, Helsinki University of Technology, Espoo, Finland.

Barba, C., J. E. Deaton, T. Santarelli, B. Knerr, M. Singer, and J. Belanich. 2006. Virtual environment composable training for operational readiness (VECTOR). In *Proceedings of the 25th Army Science Conference, 'Transformational Army Science and Technology—Charting the Future of S&T for the Soldier.'* Held Nov 27–30, Orlando, FL: Tech Science Press.

Basdogan, C., S. D. Laycock, A. M. Day, V. Patoglu, and R. B. Gillespie. 2008. 3-DoF haptic rendering. In *Haptic rendering*, ed. M. C. Lin and M. Otaduy, 311–31. Wellesley, MA: A K Peters.

Basdogan, C., and B. Loftin. 2008. Multimodal display systems: Haptic, olfactory, gustatory, and vestibular. In *The Handbook of Virtual Environment Training: Understanding, Predicting and Implementing Effective Training Solutions for Accelerated and Experiential Learning*, ed. D. Schmorrow, J. Cohn, and D. Nicholson, Vol. 2, 116–35. Westport, CN: Praeger Security International.

Begault, D. 1994. *3-D Sound for Virtual Reality and Multimedia*. Boston: Academic Press.

Biggs, S. J., and M. A. Srinivasan. 2002. Haptic interfaces. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 93–116. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Bordnick, P. S., A. Traylor, H. L. Copp, K. M. Graap, B. Carter, M. Ferrer, and A. P. Walton. 2008. Assessing reactivity to virtual reality alcohol based cues. *Addict Behav* 33:743–56.

Bordnick, P. S., A. C. Traylor, K. M. Graap, H. L. Copp, and J. Brooks. 2005. Virtual reality cue reactivity assessment: A case study in a teen smoker. *Appl Psychophysiol Biofeedback* 30(3):187–93.

Botella, C., A. García-Palacios, H. Villa, R. M. Baños, S. Quero, M. Alcañiz, et al. 2007. Virtual reality exposure in the treatment of panic disorder and agoraphobia: A controlled study. *Clin Psychol Psychother* 14:164–75.

Bowman, D., J. Gabbard, and D. A. Hix. 2002. Survey of usability evaluation in virtual environments: Classification and comparison of methods. *Presence: Teleoperators and Virtual Environ* 11(4):404–24.

Brashers-Krug, T., R. Shadmehr, and E. Bizzi, 1996. Consolidation in human motor memory. *Nature* 382(6588):252–5.

Brooks, B. M., and F. D. Rose. 2003. The use of virtual reality in memory rehabilitation: Current findings and future directions. *Neurorehabilitation* 18(2):147–57.

Brooks, B. M., F. D. Rose, E. A. Potter, S. Jayawardena, and A. Morling. 2004. Assessing stroke patients' prospective memory using virtual reality. *Brain Inj* 18(4):391–401.

Bruder, G., F. Steinicke, and K. H. Hinrichs. 2009. Arch-explore: A natural user interface for immersive architectural walkthroughs. In *Proceedings of IEEE Symposium on 3D User Interfaces (3DUI)*, 75–82, March 14–15, Lafayette, LA. Los Alamitos, CA: IEEE Press.

Bryanton, C., J. Bossé, M. Brien, J. McLean, A. McCormick, and H. Sveistrup. 2006. Feasibility, motivation, and selective motor control: Virtual reality compared to conventional home exercise in children with cerebral palsy. *CyberPsychol Behav* 9:123–8.

Bullinger, H.-J., R. Breining, and M. Braun. 2001. Virtual reality for industrial engineering: Applications for immersive virtual environments. In *Handbook of Industrial Engineering: Technology and Operations Management*, 3rd ed., ed. G. Salvendy, 2496–520. New York: Wiley.

Bungert, C. 2007. HMD/headset/VR-helmet comparison chart. http://www.stereo3d.com/hmd.htm (accessed May 22, 2010).

Burdea, G. C., and P. Coiffet, 2003. Virtual Reality Technology (2nd ed.). Hoboken, NJ: Wiley.

Burigat, S., and L. Chittaro. 2007. Navigation in 3D virtual environments: Effects of user experience and location-pointing navigation aids. *Int J Hum Comput Stud* 65(11):945–58.

Butler, R. A. 1987. An analysis of the monaural displacement of sound in space. *Percept Psychophys* 41:1–7.

Calvert, S. L. 2002. The social impact of virtual environment technology. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 663–80. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Calvert, S. L., and S. L. Tan. 1994. Impact of virtual reality on young adult's physiological arousal and aggressive thoughts: Interaction versus observation. *J Appl Dev Psychol* 15:125–39.

Carretta, T. R., and M. J. Ree. 1993. Basic attributes test (BAT): Psychometric equating of a computer-based test. *Int J Aviat Psychol* 3:189–201.

Carretta, T. R., and M. J. Ree. 1995. Air Force Officer Qualifying Test validity for predicting pilot training performance. *J Business Psychol* 9:379–88.

Carroll, M., S. Fuchs, A. Carpenter, K. Hale, R. G. Abbott, and A. Bolton. 2010. Development of an autodiagnostic adaptive precision trainer for decision making (ADAPT-DM). *Int Test Eval J* 31(2):247–63.

Carroll, M., S. Fuchs, K. Hale, B. Dargue, B. Buck. 2010. Advanced training evaluation system: Leveraging neuro-physiological measurement to individualize training. In *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) Annual Meeting*. Orlando, FL. Arlington, VA: NTSA.

Carson, D. 2000a. Environmental storytelling, part 1: Creating immersive 3D worlds using lessons learned from the theme park industry. http://www.gamasutra.com/features/20000301/carson_pfv.htm (accessed May 24, 2010).

Carson, D. 2000b. Environmental storytelling, part 2: Bringing theme park environment design techniques lessons to the virtual world. http://www.gamasutra.com/features/20000405/carson_pfv.htm (accessed May 24, 2010).

CDC. 2010. CDC Virtual World Requirements and Best Practices. http://www.cdc.gov/SocialMedia/Tools/guidelines/pdf/virtualworld.pdf (accessed June 1, 2010).

Chen, J. L., and K. M. Stanney. 1999. A theoretical model of wayfinding in virtual environments: Proposed strategies for navigational aiding. *Presence: Teleoperators Virtual Environ* 8(6):671–85.

Chua, P. T., R. Crivella, B. Daly, N. Hu, R. Schaaf, D. Ventura, et al. 2003. Training for physical tasks in virtual environments: Tai Chi. In *Proceedings of IEEE Virtual Reality Conference 2003*, 87–97, March 22–26, 2003. Los Angeles, CA. Alamitos, CA: IEEE Press.

Coelho, C. M., J. A. Santos, J. Silvério, and C. F. Silva. 2006. Virtual reality and acrophobia: One-year follow-up and case study. *Cyberpsychol Behav* 9:336–41.

Cohen, M. 1992. Integrating graphic and audio windows. *Presence: Teleoperators Virtual Environ* 1(4):468–81.

Cohn, J., P. DiZio, and J. R. Lackner. 2000. Reaching during virtual rotation: Context-specific compensation for expected Coriolis forces. *J Neurophysiol* 83(6):3230–40.

Cote, S., and S. Bouchard. 2005. Documenting the efficacy of virtual reality exposure with psychophysiological and information processing measures. *Appl Psychophysiol Biofeedback* 30:217–32.

Cruz-Neira, C., D. J. Sandin, and T. A. DeFanti. 1993. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. *ACM Comput Graph* 27(2):135–42.

Dahlquist, L. M., K. E. Weiss, E. F. Law, S. Soumitri, L. J. Herbert, S. B. Horn, K. Wohlheiter, and C. S. Ackerman. 2010. Effects of videogame distraction and a virtual reality type head-mounted display helmet on cold pressor pain in young elementary school-aged children. *J Pediat Psychol* 35(6):617–25.

Darken, R. P., and B. Peterson. 2002. Spatial orientation, wayfinding, and representation. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 493–518. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Deaton, J. E., C. Barba, T. Santarelli, L. Rosenzweig, V. Souders, C. McCollum, et al. 2005. Virtual environment cultural training for operational readiness (VECTOR). *Virtual Real* 8:156–67.

Deutsch, J. E., and A. Mirelman. 2007. Virtual reality-based approaches to enable walking for people poststroke. *Top Stroke Rehabil* 14:45–53.

DiZio, P., and J. R. Lackner, 1995. Motor adaptation to Coriolis force perturbations of reaching movements: Endpoint but not trajectory adaptation transfers to the non-exposed arm. *J Neurophysiol* 74(4):1787–92.

Draper, M. H., J. D. Prothero, and E. S. Viirre. 1997. Physiological adaptations to virtual interfaces: Results of initial explorations. In *Proceedings of the Human Factors & Ergonomics Society 41st Annual Meeting*, 1393. Santa Monica, CA: Human Factors & Ergonomics Society.

Durlach, B. N. I., and A. S. Mavor. 1995. *Virtual Reality: Scientific and Technological Challenges*. Washington, DC: National Academy Press.

Eastgate, R. M., G. D. Griffiths, P. E. Waddingham, A. D. Moody, T. K. H. Butler, S. V. Cobb, et al. 2006. Modified virtual reality for treatment of amblyopia. *Eye* 20:370–4.

Emmelkamp, P. M. G., M. Bruynzeel, L. Drost, and C. A. P. G. van der Mast. 2001. Virtual reality treatment in acrophobia: A comparison with exposure in vivo. *Cyberpsychol Behav* 4(3):335–9.

Erol, A., G. Bebis, M. Nicolescu, D. Boyle, and X. Twombly. 2007. Vision-based hand pose estimation: A review. *Comput Vis Image Underst* 108:52–73.

Ersner-Hershfield, H., J. Bailenson, and L. L. Carstensen. 2008. Feeling more connected to your future self: Using immersive virtual reality to increase retirement saving. In *Poster presented at the Association for Psychological Science Annual Convention*, Chicago, IL.

Favalora, G. E. 2005. Volumetric 3D displays and application infrastructure. *Computer* 38(8):37–44.

Figueroa, P., W. F. Bischof, P. Boulanger, and H. J. Hoover. 2005. Efficient comparison of platform alternatives in interactive virtual reality applications. *Int J Human-Computer Studies*, 62(1):73–103.

Figueroa, P., Bischof, W. F., Boulanger, P., and Hoover, H. J. 2005. Efficient comparison of platform alternatives in interactive virtual reality applications. *Int J Human-Comp Stud* 62(1):73–103.

Fouad, H. 2004. Ambient synthesis with random sound fields. In *Audio Anecdotes: Tools, Tips, and Techniques for Digital Audio*, ed. K. Greenebaum. Natick, MA: A K Peters.

Fouad, H., and J. Ballas. 2000. An extensible toolkit for creating virtual sonic environments. In *Paper presented at the International Conference on Auditory Displays, ICAD 2000*, Atlanta, GA.

Fox, J., D. Arena, and J. N. Bailenson. 2009. Virtual reality: A survival guide for the social scientist. *J Media Psychol* 21(3):95–113.

Fox, J., and J. N. Bailenson. 2009. Virtual self-modeling: The effects of vicarious reinforcement and identification on exercise behaviors. *Media Psychol* 12:1–25.

Fox, J., J. N. Bailenson, and J. Binney. 2009. Virtual experiences, physical behaviors: The effect of presence on imitation of an eating avatar. *Presence: Teleoperators Virtual Environ* 18(4):294–303.

Foxlin, E. 2002. Motion tracking requirements and technologies. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 163–210. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Fried, M. P., B. Sadoughi, M. J. Gibber, J. B. Jacobs, R. A. Lebowitz, D. A. Ross, et al. 2010. From virtual reality to the operating room: The endoscopic sinus surgery simulator experiment. *Otolaryngol Head Neck Surg* 142(2):202–7.

Fuchs, S., M. Johnston, K. S. Hale, and P. Axelsson. 2008. Results from pilot testing of a system for tactile reception of advanced patterns (STRAP). In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Held September 22–26, New York, NY. Thousand Oaks, CA: Sage Publications.

Gabbard, J. L., D. Hix, and J. E. Swan II. 1999. User-centered design and evaluation of virtual environments. *IEEE Comput Graph Appl* 19(6):51–9.

Gold, J. I., Belmont, K. A., and D. A. Thomas. 2007. The neurobiology of virtual reality pain attenuation. *Cyberpsychol Behav* 10(4):536–544.

Gopher, D., M. Weil, and M. Bareket, 1994 Transfer of skill from a computer game trainer to flight. *Human Factors,* 36(3), 387–405.

Grantcharov, T. P., V. B. Kristiansen, J. Bendix. L. Bardram, J. Rosenberg, and P. Funch-Jensen. 2004. Randomized clinical trial of virtual reality simulation for laparoscopic skills training. *Br J Surg* 91(2):146–50.

Gregg, L., and N. Tarrier. 2007. Virtual reality in mental health: A review of the literature. *Soc Psychiatry Psychiatr Epidemiol* 42:343–54.

Gress, W., and B. Willkomm. 1996. Simulator-based test systems as a measure to improve the prognostic value of aircrew selection. Selection and Training Advances in Aviation: Advisory Group for Aerospace Reasearch and Development Conference Proceedings, Prague, Czech Republic 588, 15-1–15-4.

Gross, M., S. Wurmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, et al. 2003. Blue-C: A spatially immersive display and 3D video portal for telepresence. *ACM Trans Graph* 22(3):819–27.

Gutierrez-Maldonado, J., M. Ferrer-Garcia, A. Caqueo-Urizar, and A. Letosa-Porta. 2006. Assessment of emotional reactivity produced by exposure to virtual environments in patients with eating disorders. *Cyberpsychol Behav* 9(5):507–13.

Gutierrez-Osuna, R. 2004. Olfactory interaction. In *Berkshire Encyclopedia of Human-Computer Interaction*, ed. W. S. Bainbride, 507–11. Great Barrington, MA: Berkshire Publishing.

Hale, K. S., and K. M. Stanney. 2004. Deriving haptic design guidelines from human physiological, psychophysical, and neurological foundation. *IEEE Comput Graph Appl* 24(2):33–9.

Hale, K. S., K. M. Stanney, L. M. Milham, M. A. Bell-Carroll, and D. L. Jones. 2009. Multimodal sensory information requirements for enhancing situation awareness and training effectiveness. *Theor Issues Ergon Sci* 10(3):245–66.

Harris, S. R., R. L. Kemmerling, and M. M. North. 2002. Brief virtual reality therapy for public speaking anxiety. *Cyberpsychol Behav* 5:543–50.

Hassinger, J. P., E. J. Dozois, J. D. Holubar, W. Pawlina, R. Pendlimari, J. L. Fidler, D. R. Holmes, and R. A. Robb. 2010.Virtual pelvic anatomy simulator improved medical student comprehension of pelvic anatomy. *FASEB J* 24:825.3.

Hayward, V. 2008. A brief taxonomy of tactile illusions and demonstrations that can be done in a hardware store. *Brain Res Bull* 75(6):742–52.

Hetherington, I. L. 2007. PocketSUMMIT: Small footprint continuous speech recognition. In *Proceedings of International Conference on Spoken Language Processing (Interspeech 2007—CSLP)*, 1465–8. Held August 27–31, Antwerp, Belgium. Los Alamitos, CA: IEEE Press.

Hettinger, L. J. 2002. Illusory self-motion in virtual environments. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 471–92. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Heylen, D., I. Van Es, A. Nijholt, and B. Van Dijk. 2008. Chapter 1: Controlling the gaze of conversational agents. http://en.scientificcommons.org/43376669 (accessed June 2, 2010).

Hill, R. W. Jr., J. Gratch, S. Marsella, J. Rickel, W. Swartout, and D. Traum. 2003. Virtual humans in the mission rehearsal exercise system. *Künstliche Intelligenz* 17:5–12.

Hix, D., and J. L. Gabbard. 2002.Usability engineering of virtual environments. In K. M. Stanney (Ed.), *Handbook of virtual environments: Design, implementation,* and applications pp. 681–699. Mahwah, NJ: Lawrence Erlbaum Associates.

Hoffman, H. G., D. R. Patterson, E. Seibel, M. Soltani, L. Jewett-Leahy, and S. R. Sharar. 2008. Virtual reality pain control during burn wound debridement in the Hydrotank. *Clinical Journal of Pain* 24(4): 299–304.

Holden, M. 2005. Virtual environments for motor rehabilitation: A review. *CyberPsychol Behav* 8(3):187–211.

Hollerbach, J. 2002. Locomotion interfaces. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 239–54. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Holmes, D. W., J. R. Williams, and P. Tilke. 2010. An events based algorithm for distributing concurrent tasks on multi-core architectures. *Comput Phys Commun* 181(2):341–54.

Huggins-Daines, D., M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky. 2006. PocketSphinx: a free real-time continuous speech recognition system for handheld devices. In *Proceedings of 31st International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2006)*, 185–8. Held May 14–19, Toulouse, France. Los Alamitos, CA: IEEE Press.

Isdale, J. 2000. Motion platform systems. VR News: April Tech Review. http://vr.isdale.com/vrTechReviews/MotionLinks_2000.html (accessed May 21, 2010).

Isdale, J., C. Fencott, M. Heim, and L. Daly. 2002. Content design for virtual environments. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 519–32. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Jedrzejewski, M. 2004. Computation of Room Acoustics on Programmable Video Hardware. Master's thesis, Polish-Japanese Institute of Information Technology, Warsaw, Poland.

Johnsen, K., R. Dickerson, A. Raij, B. Lok, J. Jackson, M. Shin, et al. 2005. Experiences in using immersive virtual characters to educate medical communication skills. In *Proceedings of 2005 IEEE Conference on Virtual Reality*, 179–86. Los Alimitos, CA: IEEE CS Press.

Johnston, M., K. Stanney, K. Hale, and R. S. Kennedy. 2010. A framework for improving situation awareness of the UAS operator through integration of tactile cues. In *Proceedings of the 3rd Applied Human Factors and Ergonomics (AHFE) International Conference 2010*. July 17–20, Miami, FL. New York: CRC Press/Taylor and Francis.

Jones, A., I. McDowall, H. Yamada, M. Bolas, and P. Debevec. 2007. Rendering an interactive 360° light field display. *ACM Trans Graph* 26(3, July): Article 40.

Jones, D., K. Stanney, and H. Fouad. 2005. An optimized spatial audio system for virtual training simulations: Design and evaluation. In *Proceedings of the International Conference on Auditory Display*, Held July 6–9, 2005, Limerick, Ireland. Published by the International Community for Auditory Display (ICAD).

Jones, L., C. A. Bowers, D. Washburn, A. Cortes, and R. Vijaya Satya. 2004. The effect of olfaction on immersion into virtual environments. In *Human Performance, Situation Awareness and Automation: Current Research and Trends*, ed. D. A. Vincenzi, M. Mouloua, and P. A. Hancock, Vol. 2, 282–285. Mahwah, NJ: Lawrence Erlbaum.

Kalawsky, R. S. 1993. *The Science of Virtual Reality and Virtual Environments*. Wokingham, England: Addison-Wesley.

Kallman, E. A. 1993. Ethical evaluation: A necessary element in virtual environment research. *Presence (Camb)* 2(2):143–6.

Karim, M. S., A. M. L. Karim, E. Ahmed, and M. Rokonuzzaman. 2003. Scene graph management for OpenGL based 3D graphics engine. In *Proceedings of the International Conference on Computer & Information Technology (ICCIT 2003)*, Vol. 1, 395–400. Los Alamitos, CA: IEEE Press.

Katz, N., H. Ring, Y. Naveh, R. Kizony, U. Feintuch, and P. L. Weiss. 2005. Interactive virtual environment training for safe street crossing of right hemisphere stroke patients with Unilateral Spatial Neglect. *Disabil Rehabil* 29(2):177–81.

Kaur, K. 1999. Designing virtual environments for usability. Unpublished doctoral dissertation, City University, London.

Kennedy, R. S., J. M. Drexler, M. B. Jones, D. E. Compton, and J. M. Ordy. 2005. Quantifying human information processing (QHIP): Can practice effects alleviate bottlenecks? In *Quantifying Human Information Processing*, ed. D. K. McBride and D. Schmorrow, 63–122. Lanham, MD: Lexington Books.

Kennedy, R. S., and J. E. Fowlkes. 1992. Simulator sickness is polygenic and polysymptomatic: Implications for research. *Int J Aviat Psychol* 2(1):23–38.

Kennedy, R. S., and A. Graybiel. 1965. *The Dial Test: A Standardized Procedure for the Experimental Production of Canal Sickness Symptomatology in a Rotating Environment* (Rep. No. 113, NSAM 930). Pensacola, FL: Naval School of Aerospace Medicine.

Kennedy, R. S., K. E. Kennedy, and K. M. Bartlett. 2002. Virtual environments and products liability. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 543–54. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Kennedy, R. S., N. E. Lane, K. S. Berbaum, and M. G. Lilienthal. 1993. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *Int J Aviat Psychol* 3(3):203–20.

Kennedy, R. S., and K. M. Stanney. 1996. Virtual reality systems and products liability. *J Med Virtual Real* 1(2):60–4.

Kennedy, R. S., K. M. Stanney, and W. P. Dunlap. 2000. Duration and exposure to virtual environments: Sickness curves during and across sessions. *Presence (Camb)* 9(5):466–75.

Kennedy, R. S., K. M. Stanney, J. M. Ordy, and W. P. Dunlap. 1997. Virtual reality effects produced by head-mounted display (HMD) on human eye-hand coordination, postural equilibrium, and symptoms of cybersickness. *Soc Neurosci Abstr* 23:772.

Kessler, G. D. 2002. Virtual environment models. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 255–76. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Kitazaki, M., S. Onimaru, and T. Sato. 2010. Vection and action are incompatible. In *Proceedings of the 2nd IEEE VR 2010 Workshop on Perceptual Illusions in Virtual Environments (PIVE 2010)*, ed. F. Steinicke and P. Willemsen, 22–23. March 21, 2010, Waltham, MA. http://pive.uni-muenster.de/paper/PIVE_proceedings2010.pdf (accessed May 31, 2010).

Knerr, B. W., R. Breaux, S. L. Goldberg, and R. A. Thurman. 2002. National defense. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 857–72. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Koltko-Rivera, M. E. 2005. The potential societal impact of virtual reality. In *Advances in virtual environments technology: Musings on design, evaluation, and applications*, K. M. Stanney and M. Zyda. In G. Salvendy (Ed.), HCI International 2005: 11th International Conference on Human-Computer Interaction [CD-ROM, Volume 9, unpaginated]. Mahwah, NJ: Erlbaum.

Konrad, J., and M. Halle. 2007. 3-D displays and signal processing—An answer to 3-D Ills?. *IEEE Signal Process Mag* 24(6):97–111.

Kuntze, M. F., R. Stoermer, R. Mager, A. Roessler, F. Mueller-Spahn, and A. H. Bullinger 2001. Immersive virtual environments in cue exposure. *Cyberpsychol and Behav* 4(4):497–501.

Lackner, J. R., and P. DiZio. 1994. Rapid adaptation to Coriolis force perturbations of arm trajectory. *J Neurophysiol* 72(1): 299–313.

Laughlin, D., M. Roper, and K. Howell. 2007. *NASA eEducation Roadmap: Research Challenges in the Design of Persistent Immersive Synthetic Environments for Education & Training*. Washington, DC: Federation of American Scientists. http://www.fas.org/programs/ltp/publications/NASA%20 eEducation%20Roadmap.pdf (accessed July 28, 2010).

Lawson, B. D., S. A. Sides, and K. A. Hickinbotham. 2002. User requirements for perceiving body acceleration. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 135–61. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Leonetti, C., and J. N. Bailenson. 2010. High-Tech view: The use of immersive virtual environments in jury trials. *Marquette Law Rev*, 93(3), 1073–1120.

Maekawa, T., Y. Itoh, K. Takamoto, K. Tamada, T. Maeda, Y. Kitamura, and F. Kishino. 2009. Tearable: haptic display that presents a sense of tearing real paper. In *Virtual Reality Software and Technology, Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, 27–30. New York, NY: ACM.

Majumder, A. 2003. A practical framework to achieve perceptually seamless multi-projector displays. Unpublished doctoral dissertation, University of North Carolina at Chapel Hill. http://www.cs.unc.edu/~welch/media/pdf/dissertation_majumder.pdf (accessed May 24, 2010).

Majumder, A., H. Zhu, H. Towles, and G. Welch. 2000. Achieving Color Uniformity Across Multi-Projector Displays. In *Proceedings of IEEE Visualization 2000*, (October 8–13). Salt Lake City, UT: IEEE Computer Science Press.

Mantovani, F., and G. Castelnuovo. 2003. Sense of presence in virtual training: Enhancing skills acquisition and transfer of knowledge through learning experience in virtual environments. In *Being there: Concepts, Effects and Measurement*, ed. G. Riva, F. Davide, and W. A. IJsselsteijn, 167–81. Amsterdam, The Netherlands: Ios Press.

Martinussen, M. 1996. Psychological measures as predictors of pilot performance: A meta-analysis. *J Aviat Psychol* 6:1–20.

Massimino, M., and T. Sheridan. 1993. Sensory substitution for force feedback in teleoperation. *Presence (Camb)* 2(4):344–52.

McCauley, M. E., and T. J. Sharkey. 1992. Cybersickness: Perception of self-motion in virtual environments. *Presence: Teleoperators Virtual Environ* 1(3):311–8.

McCauley-Bell, P. R. 2002. Ergonomics in virtual environments. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 807–26. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

McQuaide, S. C., E. J. Seibel, J. P. Kelly, B. T. Schowengerdt, and T. A. Furness III. 2003. A retinal scanning display system that produces multiple focal planes with a deformable membrane mirror. *Displays* 24(2):65–72.

Melzer, J. E., F. T. Brozoski, T. R. Letowski, T. H. Harding, and C. E. Rash. 2009. Guidelines for HMD design. In *Helmet-mounted Displays: Sensation, Perception, and Cognition Issues*, ed. C. E. Rash, M. Russo, T. Letowski, and E. Schmeisser, 805–48. Fort Rucker, AL: US Army Aeromedical Research Laboratory.

Melzer, J. E., and C. E. Rash. 2009. The potential of an interactive HMD. In *Helmet-Mounted Displays: Sensation, Perception, and Cognition Issues*, ed. C. E. Rash, M. Russo, T. Letowski, and E. Schmeisser, 877–98. Fort Rucker, AL: U.S. Army Aeromedical Research Laboratory.

Menzies, D. 2002. Scene management for modelled audio objects in interactive worlds. In *Proceedings of the 8th International Conference on Auditory Displays*, Kyoto, Japan: the International Community for Auditory Display (ICAD).

Milham, L. M., K. Hale, K. Stanney, J. Cohn, R. Darken, and J. Sullivan, 2004. When is VE training effective? A framework and two case studies. Poster presented at The 48th Annual Human Factors and Ergonomics Society Meeting pp. 2592–2595. New Orleans: LA.

Minamizawa, K., S. Kamuro, N. Kawakami, and S. Tachisuggest. 2008. A palm-worn haptic display for bimanual operations in virtual environments. In *EuroHaptics 2008*, ed. M. Ferre, 458–63. Berlin: Springer-Verlag.

Moline, J. 1995. *Virtual Environments for Health Care. White Paper for the Advanced Technology Program (ATP).* Retrieved September 15, 2006, from the National Institute of Standards and Technology website: http://www.itl.nist.gov/iaui/ovrt/projects/health/vr-envir.htm.

Morris, C. S., and R. W. Tarr, 2002, March. Templates for selecting PC-based synthetic environments for application to human performance enhancement and training. In *Proceedings of IEEE Virtual Reality 2002 Conference,* 109–115, Orlando, FL.

Mouba, J. 2009. Performance of source spatialization and source localization algorithms using conjoint models of interaural level and time cues. In *Proceedings of the 12th International Conference on Digital Audio Effects (DAFx-09)*. Held Sept 1–4, Como, Italy.

Mulgund, S., J. Stokes, M. Turieo, and M. Devine. 2002. *Human/Machine Interface Modalities for Soldier Systems Technologies (Final Report No. 71950-00)*. Cambridge, MA: TIAX, LLC.

Murray, J. H. 1997. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. New York: The Free Press.

Nakatsu, R., M. Rauterberg, and P. Vorderer. 2005. A new framework for entertainment computing: From passive to active experience. In: *4th International Conference on Entertainment Computing (ICEC 2005) Proceedings*, ed. F. Kishino, Y. Kitamura, H. Kato, and N. Nagata, 1–12. Lecture Notes in Computer Science (LNCS 3711), Berlin, Heidelberg: Springer-Verlag.

North, M. M., S. M. North, and J. R. Coble. 2002. Virtual reality therapy: An effective treatment for psychological disorders. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 1065–78. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Optale, G., S. Capodieci, P. Pinelli, D. Zara, L. Gamberini, G. Riva. 2001. Music-enhanced immersive virtual reality in the rehabilitation of memory related cognitive processes and functional abilities: A case report. *Presence (Camb)* 10(4):450–62.

Osgood, C. E. 1949. The similarity paradox in human learning: A resolution. *Psychol Rev* 56: 132–143

Parsons, T. D., T. Bowerly, J. G. Buckwalter, and A. A. Rizzo. 2007. A controlled clinical comparison of attention performance in children with ADHD in a virtual reality classroom compared to standard neuropsychological methods. *Child Neuropsychol* 13(4):363–81.

Parsons, T. D., and A. A. Rizzo. 2008. Affective outcomes of virtual reality exposure therapy for anxiety and specific phobias: A meta-analysis. *J Behav Ther Exp Psychiatry* 39:250–61.

Popescu, G. V., G. C. Burdea, and H. Trefftz. 2002. Multimodal interaction modeling. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 435–54. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Portillo-Rodríguez, O., C. A. Avizzano, M. Bergamasco, and G. Robles-De-La-Torre. 2006. Haptic rendering of sharp objects using lateral forces. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (ROMAN06)*, 431–6. Held September 6–8, Hatfield, U.K. Los Alamitos, CA: IEEE Press.

Povenmire, H. K., and S. N. Roscoe. 1973. Incremental transfer effectiveness of a ground-based general aviation trainer. *Human Factors*, 15(6):534–42.

Powers, M. B., and P. M. G. Emmelkamp. 2008. Virtual reality exposure therapy for anxiety disorders: A meta-analysis. *J Anxiety Disord* 22:561–9.

Pulkki, V. 1997. Virtual sound source positioning using vector base amplitude panning. *J Audio Eng Soc* 45(6):456–66.

Rauterberg, M. 2009. Entertainment computing, social transformation and the quantum field. In *Intelligent Technologies for Interactive Entertainment (INTETAIN 2009) Proceedings*, ed. A. Nijholt, D. Reidsma, and H. Hondorp, 1–8, Lecture Notes of the Institute for Computer Sciences (LNICST 9). Berlin, Heidelberg: Springer-Verlag.

Reeves, L. M., J. C. Lai, J. A. Larson, S. L. Oviatt, T. S. Balaji, S. Buisine, P. Collings, et al. 2004. Guidelines for multimodal user interface design. *Commun ACM* 47(1):57–9.

Reger, G. M., and G. A. Gahm. 2008. Virtual reality exposure therapy for active duty soldiers. *J Clin Psychol* 64:940–6.

Riecke, B. E., A. Väljamäe, and J. Schulte-Pelkum. 2009. Moving sounds enhance the visually-induced self-motion illusion (circular vection) in virtual reality. *ACM Trans Appl Percept* 6(2), Article 7, 7:1–7:27.

Rizzo, A., T. Bowerly, J. Buckwalter, D. Klimchuk, R. Mitura, T. D. Parsons. 2006. A virtual reality scenario for all seasons: The virtual classroom. *CNS Spectr* 11(1):35–44.

Rizzo, S., B. Rothbaum, and K. Graap. 2006. Chapter 9: Virtual reality applications for the treatment of combat-related PTSD. In *Combat Stress Injury Theory, Research, and Management*, ed. C. R. Figley and W. P. Nash, 295–329. New York, NY: Routledge.

Robles-De-La-Torre G., and V. Hayward. 2001. Force can overcome object geometry in the perception of shape through active touch. *Nature* 412(6845):445–8.

Rolland, J. P., F. A. Biocca, T. Barlow, and A. Kancherla. 1995. Quantification of adaptation to virtual-eye location in see-thru head-mounted displays. In *IEEE Virtual Reality Annual International—Symposium '95*, 56–66. Los Alimitos, CA: IEEE Computer Society Press.

Rolston, M. 2010. Your computer in 2020. In *Your Life in 2020*, ed. N. Perlroth. Forbes.com. http://www.forbes.com/2010/04/08/3d-computers-2020-technology-data-companies-10-frog.html (accessed May 25, 2010).

Romoser, M. R. E., and D. L. Fisher. 2009. The effect of active versus passive training strategies on improving older driver's scanning in intersections. *Hum Factors* 51(5):652–68.

Roscoe, S. N. 1982. Aviation Psychology. Ames, IA: Iowa State University Press.

Rothbaum, B. O., L. Hodges, S. Smith, J. H. Lee, and L. Price. 2000. A controlled study of virtual reality exposure therapy for the fear of flying. *J Consult Clin Psychol* 68:1020–26.

Rothbaum, B. O., L. Hodges, B. A. Watson, G. D. Kessler, and D. Opdyke. 1996. Virtual reality exposure therapy in the treatment of fear of flying A case report. *Behav Res Theropy* 34:477–81.

Roy, S., E. Klinger, P. Legeron, F. Lauer, I. Chemin, and P. Nugues. 2003. Definition of a VR-based protocol to treat social phobia. *Cyberpsychol Behav* 6:411–20.

Sadowski, W., and K. Stanney. 2002. Presence in virtual environments. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 791–806. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Sanders, P. 1997-September/October. Simulation Based Acquisition: An effective, affordable mechanism for fielding complex technologies. *Program Manager*, 72–7.

Satava, R., and S. B. Jones. 2002. Medical applications of virtual environments. In K. M. Stanney, ed. *Handbook of virtual environments*: *Design, implementation, and applications* 93–116. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Seeber, B. U., and H. Fastl. 2003. Subjective selection of non-individual head-related transfer function. In *Proceeding of the 2003 International Conference on Auditory Display*, 259–62. Held July 6–9, Boston University, Boston, MA: International Community for Auditory Display (ICAD).

Segal, D., and R. L. Fernandez. 2009. Teaching physician decision making in a technical age. *Virtual Mentor* 11(8):607–10.

Serenko, A., and B. Detlor. 2004. Intelligent agents as innovations. *Artif Intell Soc* 18(4):364–81.

Sheridan, T. B. 1993. My anxieties about virtual environments. *Presence: Teleoperators Virtual Environ* 2(2):141–2.

Shilling, R. D., and B. Shinn-Cunningham. 2002. Virtual auditory displays. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 65–92. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Siem, F. M., T. R. Carretta, and T. A. Mercatante. 1988. Personality, attitudes, and pilot training performance: Preliminary analysis (Tech. Report. N. AFHRL-TP-87-62). Brooks Air Force Base, TX: AFHRL, Manpower and Personnel Division.

Slater, M., and Steed, A. 2000. A virtual presence counter. *Presence: Teleoperators Virtual Environ* 9(5):413–34.

Spors, S., and J. Ahrens. 2010. Analysis and improvement of pre-equalization in 2.5-dimensional wave field synthesis. In *Proceedings of the 128th Audio Engineering Society (AES) Convention, P21—Multichannel and Spatial Audio: Part 2 Session (P21-3)*. Held May 25, London, U.K.: AES Publication.

Stanney, K. M., D. A. Graeber, and R. S. Kennedy. 2005. Virtual environment usage protocols. In *Handbook of Standards and Guidelines in Ergonomics and Human Factors*, ed. W. Karwowski, 381–98. Mahwah, NJ: Lawrence Erlbaum.

Stanney, K. M., and R. S. Kennedy. 1998. Aftereffects from virtual environment exposure: How long do they last? In *Proceedings of the 42nd Annual Human Factors and Ergonomics Society Meeting*, 1476–80. Chicago, IL. Thousand Oaks, CA: Sage Publications.

Stanney, K. M., and R. Kennedy. 2008. Simulator sickness. In *Human Factors in Simulation and Training*, ed. D. Vincenzi, J. A. Wise, M. Mouloua, and P. A. Hancock, 117–28. Mahwah, NJ: Lawrence Erlbaum Associates.

Stanney, K. M., K. Kingdon, I. Nahmens, and R. S. Kennedy. 2003. What to expect from immersive virtual environment exposure: Influences of gender, body mass index, and past experience. *Hum Factors* 45(3):504–22.

Stanney, K., C. Kokini, S. Fuchs, P. Axelsson, and C. Phillips. 2010. Auto-diagnostic adaptive precision training—human terrain (ADAPT-HT): A conceptual framework for cross-cultural skills training. In *Proceedings of the 3rd Applied Human Factors and Ergonomics (AHFE) International Conference 2010*. Held July 17–20, Miami, FL. Thousand Oaks, CA: Sage Publications.

Stanney, K. M., M. Mollaghasemi, and L. Reeves. 2000. *Development of MAUVE, the Multi-Criteria Assessment of Usability for Virtual Environments System (Final Report, Contract No. N61339-99-C-0098)*. Orlando, FL: Naval Air Warfare Center, Training Systems Division, 8/00.

Stanney, K. M., R. Mourant, and R. S. Kennedy. 1998. Human factors issues in virtual environments: A review of the literature. *Presence: Teleoperators Virtual Environ* 7(4):327–51.

Stanney, K. M., G. Salvendy, J. Deisigner, P. DiZio, S. Ellis, E. Ellison, et al. 1998. Aftereffects and sense of presence in virtual environments: Formulation of a research and development agenda (Report sponsored by the Life Sciences Division at NASA Headquarters). *Int J Hum Comput Interact* 10(2):135–87.

Stanney, K., S. Samman, L. Reeves, K. Hale, W. Buff, C. Bowers, et al. 2004. A paradigm shift in interactive computing: Deriving multimodal design principles from behavioral and neurological foundations. *Int J Hum Comput Interact* 17(2):229–57.

Stanney, K. M., D. D. Schmorrow, M. Johnston, S. Fuchs, D. Jones, K. Hale, A. Ahmad, and P. Young. 2009. Augmented cognition: An overview. In *Reviews of Human Factors and Ergonomics*, ed. F. T. Durso, Vol. 5, 195–224. Santa Monica, CA: Human Factors and Ergonomics Society.

Stanney, K. M., and M. Zyda. 2002. Virtual environments in the 21st century. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 1–14. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Steinicke, F., and P. Willemsen, ed. In *Proceedings of the 2nd IEEE VR 2010 Workshop on Perceptual Illusions in Virtual Environments (PIVE 2010)*. March 21, 2010, Waltham, MA. http://pive.uni-muenster.de/paper/PIVE_proceedings2010.pdf (accessed May 31, 2010). Los Alamitos, CA: IEEE Press.

Sterling, G. C., L. E. Magee, and P. Wallace. 2000. Virtual reality training—A consideration for Australian helicopter training needs? In *Paper presented at Simulation Technology and Training (SimTecT2000)*, March 2000. Sydney, Australia: Simulation Industry Association of Australia.

Stoffregen, T., B. G. Bardy, L. J. Smart, and R. Pagulayan, 2003. On the nature and evaluation of fidelity in virtual environments. In *Virtual and adaptive environments: Applications*, implications, and human performance issues. L. J. Hettinger and M. W. Haas, eds. 111–128, Mahwah, NJ: Lawrence Erlbaum Associates.

Stone, R. 2002. Applications of virtual environments: An overview. In Handbook of virtual environments: *Design, implemen-tation, and applications*. K. M. Stanney, ed. 827–856. Mahwah: NJ: Lawrence Erlbaum Associates.

Storms, R. L. 2002. Auditory-visual cross-modality interaction and illusions. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 455–70. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Strickland, D., L. Hodges, M. North, and S. Weghorst. 1997. Overcoming phobias by virtual exposure. *Commun ACM* 40(8):34–9.

Sullivan, A. 2004. DepthCube solid-state 3D volumetric display. In *Proceedings of the SPIE Stereoscopic Displays and Virtual Reality Systems*, Vol. 5291, 279–84. SPIE Press.

Suma, E. A., S. Clark, S. L. Finkelstein, and Z. Wartell. 2010. Leveraging change blindness for walking in virtual environments. In *Proceedings of the 2nd IEEE VR 2010 Workshop on Perceptual Illusions in Virtual Environments (PIVE 2010)*, ed. F. Steinicke and P. Willemsen, 10. Held March 21, 2010, Waltham, MA. http://pive.uni-muenster.de/paper/PIVE_proceedings2010.pdf (accessed May 31, 2010). Los Alamitos, CA: IEEE Press.

Swartz, K. O. 2003. Virtual environment usability assessment methods based on a framework of usability characteristics. Unpublished master's thesis. Virginia Polytechnic Institute and State University, Blacksburg.

Tharp, V., M. Burns, and H. Moskowitz. 1981. *Development and Field Test of Psychophysical Tests for DWI Arrest (Department of Transportation Final Report, ODT HS 805 864)*. Washington, DC: DOT.

Thorndike, E. L., and R. S. Woodworth. 1901. The influence of improvement of one mental function upon the efficiency of the other functions. *Physiol Rev* 8(3):247–61.

Tippett, W. J., J -H. Lee, K. K. Zakzanis, S. E. Black, R. Mraz, and S. J. Graham. 2009. Visually navigating a virtual world with real-world impairments: A study of visually and spatially guided performance in individuals with mild cognitive impairments. *J Clin Exp Neuropsychol* 31(4):447–54.

Turk, M. 2002. Gesture recognition. In *Handbook of Virtual Environments: Design, Implementation, and Applications*, ed. K. M. Stanney, 223–38. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Ultimate 3D Links. 2010. Commercial 3D software. http://www.3dlinks.com/links.cfm?categoryid=1&subcategoryid=1# (accessed May 25, 2010).

Vertanen, K., and P. O. Kristensson. 2009. Parakeet: A continuous speech recognition system for mobile touch-screen devices. In *Proceedings of the 13th International Conference on Intelligent user Interfaces*, 237–46. February 9–11, Sanibel Island, FL: ACM.

Vince, J. 2004. *Introduction to Virtual Reality*, 2nd ed. Berlin: Springer-Verlag.

Washburn, D. A., and L. M. Jones. 2004. Could olfacatory displays improve data visualization? *Comput Sci Eng* 6(6):80–3.

Washburn, D. A., L. M. Jones, R. V. Satya, C. A. Bowers, and A. Cortes. 2003. Olfactory use in virtual environment training. *Model Simul* 2(3):19–25.

Welch, R. B. 1978. *Perceptual Modification: Adapting to Altered Sensory Environments*. New York: Academic Press.

Welch, R. B. 1997. The presence of aftereffects. In *Design of Computing Systems: Cognitive Considerations*, ed. G. Salvendy, M. Smith, and R. Koubek, 273–76. Amsterdam, The Netherlands: Elsevier Science Publishers, San Francisco, CA, August 24–29.

Wiecha, J., R. Heyden, E. Sternthal, and M. Merialdi. 2010. Learning in a virtual world: Experience with using Second Life for medical education. *J Med Internet Res* 12(1):e1. http://www.jmir.org/2010/1/e1/ (accessed May 20, 2010).

Wickens, C. D., and J. G. Hollands. 2000. Engineering psychology and human performance (3rd ed). New Jersey: Prentice Hall.

Witmer, B., and M. Singer. 1998. Measuring presence in virtual environments: A Presence Questionnaire. *Presence: Teleoperators Virtual Environ* 7(3):225–40.

Yee, N., J. N. Bailenson, and N. Ducheneaut. 2009. The proteus effect: Implications of transformed digital self-representation on online and offline behavior. *Commun Res* 36(2):285–312.

Yeh, S.-C., T. D. Parsons, M. McLaughlin, A. A. Rizzo. 2007. Virtual reality upper extremity motor training for post-stroke rehabilitation. *J Int Neuropsychol Soc* 13(Suppl S1):58.

Zittel, R. C. 2001. Summer. The reality of simulation-based acquisition-And an example of U.S. military implementation. *Acquisition Review Quarterly*, 121–32.

Zotkin, D. N., R. Duraiswami, E. Grassi, and N. A. Gumerov. 2006. Fast head related transfer function measurement via reciprocity. *J Acoust Soc Am* 120(4):2202–15.

29 Privacy, Security, and Trust

Human–Computer Interaction Challenges and Opportunities at Their Intersection

John Karat, Clare-Marie Karat, and Carolyn Brodie

CONTENTS

29.1 INTRODUCTION

In this chapter, we discuss three topics—privacy, security, and trust—which all have a variety of meanings in human contexts in a multidisciplinary view of the world. This presents a challenge in talking about any of them in isolation, and also in addressing questions of deciding what human–computer interaction (HCI) research might be relevant to each. To help set the context for this chapter, we will make use of a unifying scenario. As the discussion progresses, we will identify HCI research views that have been most useful in developing a productive research agenda for the field. The scenario below is drawn from our work in bridging the gap between business and technology perspectives in the area. As in our research work, we use the scenario to highlight issues rather than to offer an accurate picture of an existing situation or details of an idealized proposed solution.

29.1.1 Medical Scenario as Context for Privacy, Security, and Trust Issues

Patient Moves to New City, Selects and Sees New Doctor, Later Goes to Hospital for Tests, Receives Referral to Specialist, Communicates with Family, Friends, and Business Associates in United States and Europe

Mary Simpson and her husband have moved from Boston to New York and started a new job in the financial sector. In her new job, Mary collaborates with people in New York and colleagues in Europe. Before leaving Boston, she requested and obtained a copy of her medical records, containing both paper records and pointers to electronic information from her physician. As part of getting settled into the new community, she selects a new doctor based on referrals from people at work. The group practice that the doctor is part of is particularly forward-looking and thorough. The group practice has created a privacy policy and formulated procedures for collection, use, and storage of their patients' medical data. To do this, they have reviewed the Health Insurance Portability and Accountability Act guidelines and written a policy and set of procedures to cover the different situations that arise. Although some data are kept in hard copy files, much of the patient record is kept electronically in a commercial patient record system called MedicalFiles.

The group practice's privacy policies cover their operational rules for disclosing information to healthcare professionals within the practice and in other organizations who are coordinating in the care of patients in the group practice, disclosure of information to insurance companies and national health center organizations, and the privacy policy preferences that the patients can set up for allowing access to the patient view of the medical file called MyMedicalFile. Similarly, the group practice has a set of security policies that it follows for both the hardcopy and electronic patient data. The database administrator for the group practice implements both the privacy and security policies for the MyMedicalFile electronic patient record system and coordinates with the office manager on the implementation of privacy and security policies for hardcopy data within the group practice. The database administrator uses commercial security and privacy software with the MedicalFile application. The security software protects against unauthorized access. The privacy software controls access to patient data for different purposes and logs the access information. The doctor's office has data retention policies for individually identifiable health information (IIHI) data and conducts periodic audits to ensure the accuracy of IIHI data in medical summaries and billings to insurance companies. The retention policy covers patients who move away or have died.

The MedicalFile application is hosted on the web server and provides a MyMedicalFile view for patients to view information about their medications, their medical summaries, and their laboratory test reports, and also to make appointments. Mary makes an appointment to see the doctor. At the time of the appointment, she brings in her previous medical records including laboratory reports, doctor's summaries, and hard copies of reports such as mammograms or x-rays, with pointers to electronic records. Mary's data are collected by the doctor's staff and with the new information she provides during the initial visit with the doctor, the information forms her patient record file in this group practice. Mary's patient record is a combination of paper, x-ray, and electronic IIHI. Mary is informed about the privacy policy regarding her data and consents to it. She is also informed about how to access her online data in the patient record system. Before Mary leaves the office, the office staff submits her bill to her insurance company.

A few months later Mary becomes ill and wants to see the doctor regarding her illness. She goes online to make an appointment to see the doctor. She visits the doctor, and he orders a number of lab tests. Some of these tests are done in the doctor's medical office by the nurse on staff. Others are scheduled by the nurse to be done at a nearby hospital with which the doctor is affiliated. Mary completes the tests to be done in the medical office at the end of the visit. Again, Mary's bill for seeing the doctor is completed and submitted to her insurance company before she leaves the doctor's office.

Mary visits the hospital on an outpatient basis and completes the tests. Her insurance is billed by the hospital. A few days later, her test results are forwarded by the hospital to the patient record system maintained by her doctor's practice. The data are encrypted during this process for security and privacy reasons. Mary is able to see her test results through her MyMedicalFile view of her online patient record. No one else in Mary's family can view her results in the online medical record as she has not authorized her family through the policy preferences she controls in the MyMedicalFile system to have access to the data. Because she trusts him and for convenience, Mary decides that she will change her profile to allow her husband access to her test results in the future. Any unauthorized attempts to view Mary's records are noted and logged.

Mary is an active user of a social networking tool and communicates with her friends, family, and colleagues through the tool about her health issues. She has previously set up different groups with different levels of access to her information. She uses these groups to ensure that only the personal and professional information she wants to share is disclosed to personal friends, family, and business associates in the United States and in Europe. During this ongoing medical situation, she provides updates to her family, friends, and colleagues as appropriate. It is helpful for her to get the information out to people in this way. Although Mary phones her closest family member and friends, the use of the social networking tool with the proper security and privacy profiles allows her to let all the appropriate people know her situation without making a large number of phone calls to people with much different geography.

In reviewing the results, Mary's primary physician determines that the tests have revealed that Mary has cancer. There are a number of possible treatment alternatives for the type of cancer that Mary has. Mary's doctor wants to have Mary see a specialist, an oncologist, for another opinion about how to proceed based on the test results. He talks with Mary about this during an office visit to review her test results with her. Mary agrees to see the specialist before a treatment plan is defined. Before going to see the specialist, Mary searches the web for information about her medical condition.

Mary's doctor completes the referral to the specialist, and the staff at her doctor's office makes the appointment for Mary to see the specialist. As part of the privacy policy of the group practice, the specialist is granted access to review Mary's most recent test results in the online patient medical record as the doctor is part of Mary's healthcare team now. Mary's primary physician also has his staff forward specific elements of her data in an anonymized manner to a national database conducting medical research on the disease she has.

29.1.2 Policy as a Central Issue in Security, Privacy, and Trust

It should be clear from the scenario above that security, privacy, and trust are influenced by rules that are specific to a context. When Mary allows her husband to see her medical history, she intends to create a rule that gives him access to her records. She trusts that he will use the information appropriately. She might even trust that the system would not allow her records to be tampered with by people not authorized by the doctors to do so. We will consider these types of rules—rules that guide the operation of various systems—to be *policies*. Broadly speaking, policies might be "built into" systems or they might be "set" by system administrators or even end users. When they are built in, they become difficult to change or modify. This can be appropriate when the policies are fixed across time and user population, or it can be problematic if the context calls for flexibility. Such policies have a particularly important role in security and privacy systems. Because the context can be quite dynamic (different people can view sharing their information differently, and new intrusion threats can develop that security systems need to respond to), policies are generally authored by users with some intention. Thus, policy management (which includes how policies are authored, modified, evaluated, and implemented) becomes an important issue to be considered within security, privacy, and trust contexts.

29.1.3 Relationships among Privacy, Security, and Trust

What do we mean when we say that we trust a computer system? Although trust is a complex concept with many different meanings in the behavioral science literature, it can be seen as having a fairly specific meaning when applied to a person's interaction with a technology system. Trust in a system is the extent to which a system behaves as the user expects. Put another way, it is a measure of how much users believe that the system will do what they ask it to do without causing any harm. This view suggests that people can enter an interaction with a system with some ideas about what the interaction will involve, and that their trust in the system can either increase or decrease with experience. Trust becomes important, particularly for people or organizations that rely on people wanting to use their websites or products, because without trust in a system, people will find other ways to conduct their business. Parts of the expectations people have about systems are related to the intended function or purpose of the system. They trust that automatic teller machines (ATMs) will correctly distribute money and debit their accounts appropriately. They expect that merchandise ordered through a website will arrive safely at their homes. But, there are also expectations about the handling of information they exchange with the system. They assume that transactions are secure—that they know who they are dealing with and that information only reaches the intended target. They also assume that their privacy is respected—that promises the owners of the system make about how the information will be used or shared are kept. This intersection of trust, security, and privacy is becoming increasingly important as information technology (IT) becomes a more pervasive part of our lives.

There can be many aspects and influences on the trust placed by a user in a computer system. We will not try to address them all in this chapter, but will focus on two important contributing factors associated with the "risks of harm" one might have in interacting with a system. Generally speaking, when someone interacts with a system, they assume that doing so will not cause them any harm. They expect, that is they trust, that the interaction is truthful and without hidden consequences. If they interact with a website, they assume that it represents communication with a specific person or organization. If they disclose information to the site, they assume that the disclosure will only reach the intended target and that only appropriate use will be made of the information.

We will consider security as the degree to which a system can protect information it contains. There are many factors associated with security in systems that interact with humans. Perhaps the most important of these is authentication—how a system and a user can be confident of each other's identity.

If the user and system each have confidence that they know the identity of the other, we generally assume that neither will attempt to do harm to the other. Although research in system security is often aimed at minimizing risks associated with malicious attacks on systems, for HCI, we will focus on the trade-offs between rigorous authentication and ease of interaction.

The concept of privacy extends beyond security to examine how well the use of information that the system acquires about a user conforms to the explicit or implicit assumptions regarding that use associated with the personal information (PI). There is an important distinction that we would like to draw when discussing privacy from an HCI view. In general, studies have looked at privacy from two different perspectives. From an end-user perspective, privacy can be considered as restricting access to their PI, or it can be viewed as controlling use of PI. In the former, the user expresses a “wish to be left alone”; in the latter the wish is to “use my information according to expressed wishes.” Although much has been said about the end of privacy in the pervasive computing world—meaning that so much is known about each of us that it is futile to worry about privacy—this view is an attempt to highlight the difficulty people face in trying to remain anonymous (the first consideration above). Although electronic surveillance is increasingly common, this does not mean that people have to give up their rights to control the use of information collected about them. It is our assumption that collection of more and more data about us is a trend that will continue. But we also assume that legislation will support people’s rights to ensure that appropriate use is made of data collected. In considering privacy, we will generally assume that security in a system is adequate. Thus, we will view data protection failures associated with unauthorized access to information to be security failures or breaches, and those associated with noncompliance to stated privacy policies to be privacy breaches. We will explore the privacy issues in more detail below. For the purposes of this chapter, a simple but useful definition of privacy is

> The ability of individuals to control the terms under which their PI is acquired and used (Culnan 2000, p. 21).

In summary, security involves technology to ensure that information is appropriately protected. It involves users in that security features such as passwords for access control or encryption to prevent unintended disclosure, often place requirements on users to function correctly. Password schemes must be hard to guess but should be easy to remember. Encryption mechanisms must prevent unintended decryption but be transparent (or nearly so) to intended senders and recipients. Privacy involves mechanisms to support compliance with some basic principles. Basically, these suggest that people should be informed about information collection, told in advance what will be done with their information, and given a reasonable opportunity to approve of such use of information. Trust is seen as increasing when it is perceived that security and privacy are provided for. Without trust, it is perceived that people will be less likely to use systems.

As a part of our team’s line of research in these areas, we identified connections between the concepts of privacy, security, and trust as they applied to interaction with organizations that collect PI. This work (Karat et al. 2005) used a contextual design method, which enabled us to identify themes presented in data and to understand the relationships among them. One theme involved the relationship of privacy to other concepts. Many of the participants we talked with discussed how privacy related to other concepts such as security, personalization, trust, and education. Figure 29.1 shows how each of these concepts relates to privacy. Good security enables and is a building block for privacy, and the interviewees noted that as they focus on managing privacy in their organizations, they have found ways to enhance their security as well. The trust that data subjects have in an organization is paramount, and providing privacy protection is a critical means of ensuring that trust. Privacy education is important both for the organization and their external users. Many interviewees stated that privacy education of their employees is an ongoing priority now and that they believe the privacy of data subjects’ PI cannot be effectively protected until employees (the data users) believe it is important and follow through in their daily actions. Many have initiated training programs and have frequently asked questions (FAQ) documentation available for employees to access to understand how to handle PI correctly. The interviewees in the current research study stated that their customers would only provide PI on the condition that it would be protected and not misused and this is a critical element in personalization. Privacy is woven into organizations’ business processes. Interviewees stated that they are reexamining their business processes to protect the PI of their employees, customers, constituents, and patients. They are finding redundancies in the collection of PI and are realizing financial benefits from streamlining their processes privacy management.

The privacy and security functionality of the applications that users experience must also be usable to gain the user’s trust. There are unique challenges in making these capabilities usable. We will explore these issues before discussing the research in privacy, security, and trust.

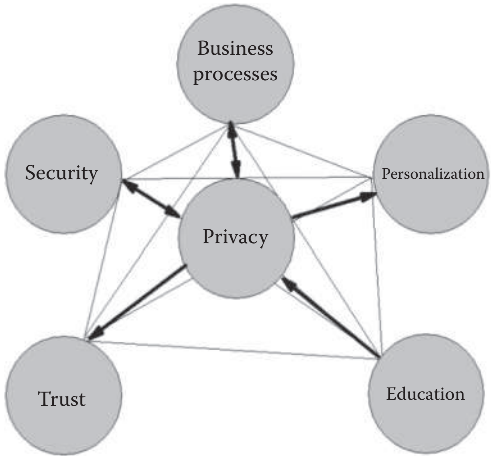


FIGURE 29.1 The relationships between privacy and other concepts.

29.2 HUMAN–COMPUTER INTERACTION AND USABILITY IN THE PRIVACY, SECURITY, AND TRUST DOMAINS

Although the core usability goals of understanding the user, their tasks, and the context of use apply across a wide range of domains, there are unique aspects of privacy and security that present challenges and opportunities when designing security and privacy functionality. First, a key issue to consider is that the use of security and privacy solutions is not the user's main goal. Users value and want security and privacy functionality and they are not likely to trust systems that do not provide them, but they regard them as only secondary to completing their primary tasks (e.g., completing an online banking transaction, ordering medications). The central conflict in the minds of many is that security mechanisms are seen as making operations harder, whereas usability is focused on making operations easier. The user would like the solutions to be as transparent as possible, but users do want to be in control and understand what is occurring. Therein lies the HCI challenge. Thus, the display of information and the interaction methods related to security and privacy solutions need to be accessible if and when desired by the user.

Second, as more of people's interactions in daily life involve the use of computing technology and sensitive information, disparate types of users must be accommodated. Security solutions in particular have historically been designed with a highly trained technical user in mind. The user community has broadened extensively as organizational business processes have come to include new types of roles and users in the security and privacy area. Many compliance and policy roles in organizations are handled by legal and business process experts who have limited technical skills. Moreover, the end-user base includes virtually everyone in the population. The functionality provided by the system for people with different roles must accommodate the skill sets of each. Security and privacy are requirements for doing business as an organization and must be done well, or the organization may lose the user as a customer or worse.

Third, the risk of the negative impact of usability problems is higher for security and privacy applications than for many other types of systems. Although complexity is at the very heart of many security and privacy solutions, from an HCI point of view that complexity is really the enemy of the success of security and privacy. If the system is so complex that the various users groups (e.g., technical users, business users, and end users) cannot understand it, then costly errors will occur. There is a saying in the HCI field that: "If the user cannot understand functionality, it doesn't exist." In the case of security and privacy, badly designed functionality may put users at more risk than if they used less sophisticated solutions. So, the increased risk of errors in this domain provides an even greater incentive to include HCI work in system research and development. The user issues in the domain provide unique technical challenges to architects to design the solutions to be simple and effective.

Fourth, organizations and their users will need to be able to easily update security and privacy solutions to accommodate frequent changes in legislation and regulations. Additionally, different domains (e.g., health care, banking, government) and geographies will have unique requirements. Systems must be designed to enable easy and effective updates to them. Although this list is not exhaustive, these challenges provide a unique focus and strong incentive to include HCI in the system lifecycle. We discuss the valuable role that HCI can play in more detail below.

29.3 VIEWS ON PRIVACY

The rapid advancement of the use of IT in industry, government, and academia makes it much easier to collect, transfer, and store PI around the world. This raises challenging questions and problems regarding the use and protection of PI (Kobsa 2002). Questions of who has what rights to information about us for what purposes become more important as we move toward a world in which it is technically possible to know just about anything about just about anyone. As stated by Adams and Sasse (1999, p. 41): "Most invasions of privacy are not intentional but due to designers' inability to anticipate how this data could be used, by whom, and how this might affect users." Deciding how we are to design privacy considerations in technology for the future includes philosophical, legal, and practical dimensions—any or all of which can be considered as within the domain of the field of HCI.

Privacy can and does mean different things to different people. This chapter primarily focuses on a view of privacy as the right of an individual to control PI use rather than as the right to individual isolation. Organizations commonly provide a description of what kind of information they will collect and how they will use it in privacy policies. In some areas (e.g., the collection and use of healthcare information in the United States or movement of PI across national boundaries in Europe), such policies can be required, though the content of the policy is not generally specified in legislation. Although there has been considerable consensus around a set of high-level privacy principles for IT, it is not likely that a single privacy policy can be created to address all information privacy needs. For example, there will likely be considerable differences in privacy legislation in different regions of the world. Similarly, organizations in different fields (e.g., healthcare, banking, government) need to tailor policies to their domains and needs. This chapter focuses on privacy policy, although privacy is not entirely about "setting rules and enforcing them" (Palen and Dourish 2003, p. 129). To implement privacy within an organization, the coordination of people, business processes, and technology is required (Karat et al. 2005). Still, we do believe that such policies are essential when interacting with technology and organizations in that they enable people to better understand the boundaries between public and private information and technology.

It is interesting to note that while privacy policies are not new to most organizations, very little has been done to implement the policies through technology (Smith 1993). There are emerging standards for privacy policies on websites (Cranor 2002), but these address machine-readable policy content without specifying how the policy might be created or implemented. The reality is that there is very little capability to have technology actually implement access and disclosure limitations that we might expect from a policy statement like: "We will not share your information with a third party without your consent." The emerging focus is on how organizations could create a wide range of policies, and how technology might enable the policies to be enforced and audited for compliance. Karat et al. (2005) focus on technology to enable usable privacy policy authoring and enforcement, rather than trying to directly address what privacy rights people should have or how to de-identify information such as video stored in systems (e.g., Senior et al. [2003]). Privacy is an important social issue and technology that can enable flexible, reliable, and verifiable privacy policy enforcement to preserve individual rights.

The situation with respect to privacy in organizations is described in a Forrester report (Hagan 2000). This research reveals a mismatch between consumer demands for privacy and enterprise practices in industry. According to this report, although customer concerns about privacy remain high, the majority of executives (58%) believe privacy issues are addressed extremely well by their companies. Most executives do not know whether their customers check the privacy policies or not and few see the need to enhance their privacy practices. Research in the Asia-Pacific region complements these results (Office of the Federal Privacy Commissioner of Australia 2000). With these results in mind, we suggest that privacy protection must extend across the network into enterprise processes, and that there is a need to audit data collection and sharing mechanisms. We agree that technology design generally does reflect concerns of society in general (Ackerman and Mainwaring 2005), and believe that we are experiencing a shift toward a greater concern for privacy in IT design. Currently, end users of social networking applications such as Facebook and search engines such as Google are pushing back on these organizations and demanding new customer defined boundaries for the use of their PI (Helft and Wortham 2010; Opsahl 2010).

Central to our view of privacy is the notion that the parties involved in information exchanges have implicit or explicit policies with regard to the use of the information. This applies both to the person whom the information is about and to the organization collecting and using the information. In the privacy literature on organizations, although some attention has been given to the generally implicit policies of end users whose PI is being collected and used (often called data subjects), the main focus is on the policies of the organization collecting the information. Smith (1993) described such organizational policies, and also noted the lack of technology in enforcing the policies. He described the rather unstructured ways in which organizations develop privacy policy—a characterization that has changed little in the nearly 20 years since his research was published in spite of the increased legislation of the past few years. Future research must address both the needs of data subjects and organizations by addressing the gap between policy and practice.

In the medical scenario described above, there are a number of privacy questions related to Mary's healthcare episode. Who can access Mary's medical data and for what purposes? Who authorizes the access? How can Mary easily and effectively manage the disclosure of information to people so that she is comfortable and in control? What steps are available to monitor and assure that Mary's privacy is being protected as intended in the health care and insurance environments she interacts with and in the personal and professional areas of her life? Section 29.3.1 will review HCI research on privacy with these questions in mind.

29.3.1 Human–Computer Interaction Research on Privacy

Rapid advances in technology are pushing us closer and closer to Mark Weiser's (1991) vision of a ubiquitous computing world. Technology has permeated every facet of our daily lives—from enabling us to shop online with a few clicks of the mouse, to recommending movies appropriate for our interests (Good et al. 1999), to helping us share documents with colleagues on the team (e.g., Horstmann and Bentley 1997), to allowing us to estimate when a collaborator may be available for communication (e.g., Begole et al. 2002), to reminding us of relevant tasks based on our current context (e.g., Dey and Abowd 2000), and so on. The benefits of these technological developments promise to improve quality and experience of life by providing people convenience and efficiency. To effectively provide these benefits, systems need to capture, store, and analyze information that may be deemed sensitive by the individuals concerned. The quantity of information varies from system to system, and the associated sensitivity can vary from individual to individual and context to context. As a result, to fully achieve their potential such systems must first overcome various social hurdles. In particular, user perceptions regarding potential violations of privacy have emerged as a key factor that affects technological acceptance and adoption (Herbsleb et al. 2002; Want et al. 1992). For example, the ambitious Total Information Awareness initiative proposed by the U.S. government was abandoned before proceeding beyond the planning stages due to its inability to satisfactorily address privacy fears raised by citizens concerned with civil liberties (Associated Press 2003). As rampant increases in viruses, worms, spam, spy-ware, and ad-ware threaten to erode user confidence, it has become increasingly important that a system empower users to appropriately manage privacy.

The motivation in studying these issues within the field of HCI lies in the interest in designing computing systems with privacy in mind at the outset. Although researchers have acknowledged the existence of privacy concerns in a wide variety of technological domains: e-mail (e.g., Bellotti 1996), e-commerce (e.g., Egelman et al. 2009; Ackerman, Cranor, and Reagle 1999), media spaces (e.g., Mantei et al. 1991), data mining (e.g., Thuraisingham 2002), homes of the future (e.g., Little, Sillence, and Briggs 2009; Meyer and Rakotonirainy 2003), designing effective solutions to address these concerns is challenging. One of the reasons is that relatively few empirical studies have been conducted with the sole aim of studying privacy issues. In addition, empirical investigations of privacy pose numerous methodological challenges. To start with, privacy has proved hard to define due to its highly nuanced and context-dependent nature. Thus, individual differences in perceptions and interpretations of privacy could lead to researcher-introduced bias. For example, in a review of 23 privacy surveys, Harper and Singleton (2001) point out that some surveys may suffer from manipulative questioning on the one hand and that unprompted surveys may reveal a low level of user privacy concerns on the other hand. It is also possible that certain privacy issues remain undetected because the researcher(s) did not recognize them as such. Moreover, cultural differences in expectations and behaviors regarding these issues tend to be quite profound (Milberg et al. 1995), making it difficult to generalize findings across cultures, or to study settings that involve individuals from multiple cultures. Differences in privacy laws in different countries could make it difficult to isolate actual intention from mere legal compliance. Further, methodologies for studying privacy may themselves be deemed too privacy invasive, causing users to deviate from normal practice and/or to withhold revealing sensitive aspects. As a result, relying on self-reported attitudes and behavior alone may not provide a valid view of normal practices (Spiekermann, Grossklags, and Berendt 2001).

With this overview of the privacy and HCI area as context, we will now examine a selection of key privacy research in two topic areas: user control of PI and organizational requirements in managing privacy. Although this is not an exhaustive review, it will illustrate the HCI issues in the area of privacy.

29.3.2 User Control of Personal Information

29.3.2.1 End-User Views of Privacy

It is important to Mary that her physician's group practice takes great care to protect the privacy of her medical data within their practice and in dealings with other medical specialists involved with her care and with insurance organizations processing claims for health care services. As an end user of the online medical file application, she has decided not to share access to her medical file with other family members except her husband. In the business and personal parts of her life, she has carefully determined what information to disclose to groups through verbal interactions and as an end user of web-based work environments and social networking services.

Researchers have conducted many studies on end-user preferences regarding privacy on the web and the HCI factors that are necessary to satisfy the user's desire for privacy. Jensen and Potts (2004) found that while most surveys of user concerns about privacy show high rates of behaviors such as reading the privacy policy or taking concrete actions to protect their privacy, informal analysis of log-file data suggests that the actual rates of these user behaviors are much lower. Jensen, Potts, and Jensen (2005) found that users place inappropriate trust in the presence of trust indicators such as the TRUSTe mark or a privacy policy on the site, assuming quality in the presence of these trust indicators rather than understanding that the level of their trust should be dependent on the content of these policies. The users were willing to divulge PI when it was not warranted. Jensen and Potts (2004). recommend that HCI professionals work to increase awareness of the privacy issues related to the policy issues associated with the trust indicators, as unscrupulous online vendors could use the trust indicators to mislead users to accept privacy policies and divulge private information that they would not otherwise willingly do. Buffett et al. (2004) have created a technique for enabling a user to compute the value of the consequences of exchanging PI on the web. The paper provides a demonstration of the effectiveness of the technique in improving a user's expected utility in a simple privacy negotiation.

Little, Sillence, and Briggs (2009) have carried out some interesting scenario-based work to investigate a variety of social impact issues, including privacy concerns for pervasive technology. They observe that developments in ubiquitous and pervasive computing herald a future in which computation is embedded into our daily lives. Such a vision raises important questions about how people, especially families, will be able to engage with and trust such systems while maintaining privacy and individual boundaries. Their work includes the development of approaches (e.g., the use of illustrative videos) and tools (e.g., a Pre-Concept Evaluation Tool for use in design and implementation of ubicomp systems) for investigating such issues. Over 300 UK citizens participated in 38 focus groups. The groups were shown videotaped scenarios depicting pervasive applications in a number of contexts, including shopping. The results covered family concerns over who controls information, who sees information, who benefits from the use of information, and who is responsible for data control. The data raises a number of important issues from a family perspective in terms of access, control, responsibility, benefit, and complexity. Also findings highlight the conflict between increased functionality and the subtle social interactions that sustain family bonds.

Cranor (2005) leads the W3C standardization work in the area of the Platform for Privacy Preferences (P3P) policies. P3P policies provide end-users information about the privacy policies of a site before they interact with it. Dr. Cranor and a team at AT&T Labs designed and developed the Privacy Bird, a browser help object that provides summary information about the agreement or lack thereof between the end-user's privacy preferences and the privacy policy of the website with which the person is considering interacting. Research continues to improve the usability of the Privacy Bird in communicating to users about website privacy policies and in capturing user privacy preferences. At the current time, there is no verification possible that organizations are operating according to their stated P3P policies. An important area for future research is the challenge of linking the P3P policy to the internal operational privacy policies and their implementation in organizations, and then enabling compliance audits of policy execution.

Peer-to-peer (P2P) systems enable users to easily share files, by downloading data simultaneously from multiple sources and sharing many different file types. Good and Krekelberg (2003) illustrate HCI problems with P2P file-sharing systems such as KaZaA. KaZaA, the most popular and widely used P2P tool in 2003, had an average of 120 million downloads worldwide and 3 million users online at any given time during the course of the study. Good et al. conducted a user study of KaZaA and found that a large proportion of users unwillingly share personal and private files, leaving them at risk of being taken advantage of through unknown exposure of PI. The majority of the users in the study were unable to correctly determine the files they were sharing. Many thought they were sharing no files when they were actually sharing all the files on their hard drive.

The area of ubiquitous computing provides many new capabilities for users to use in communicating with the outside world. Users can communicate using a number of different mobile, pervasive devices, and they have many choices in terms of managing their privacy in these new computing contexts. Researchers have conducted a series of studies in this domain. In the first study, Lederer, Mankoff, and Dey (2003) conducted a questionnaire-based study on the relative importance of the inquirer and the situation in determining a user’s decision regarding the preferred accuracy of PI disclosed through the ubiquitous device. They found that users were more likely to provide the same level of accuracy in the disclosure of PI to the same inquirer in different situations rather than to different inquirers in the same situation. In later research, Hong et al. (2004) developed a privacy risk model for designing privacy-sensitive ubiquitous computing systems and present two case studies that illustrate the use of the privacy risk model in the design of applications. The goal in the research was to refine the concepts of privacy to concrete issues that could be addressed by users using different applications. They found through the case study research that although there were many variables to manage, people did not want privacy management to be complicated or take very much time. People, on occasion, also wanted the ability to provide inaccurate or false information. Hong and Landay (2004) create a toolkit for developers called Confab that facilitates the development of privacy-sensitive ubiquitous applications. Confab includes a framework and customizable privacy mechanisms. The tool also includes extensions for managing location privacy. The goal of Confab is to enable application developers to address the range of trust and privacy requirements of end users of ubiquitous computing systems.

In terms of providing solutions for users to manage their privacy in the future, the European Union is sponsoring research on a system called PRIME (PRivacy and Identity Management for Europe). The goal of the PRIME system is to support the user in controlling the PI about himself that is disclosed to others through interactions with individuals, systems, and organizations (Pettersson et al. 2005). The Pettersson et al. (2005) paper describes three alternative user interface paradigms for privacy-enhanced identity management and illustrates how key legal privacy principles from the European Union Directives are associated with these designs. In this publication on the first year of research in this effort, the team reports some results of initial usability evaluations of mock ups of the three paradigms. The final project report was delivered in 2008 and covered research on graphical user interface, authorization models, cryptography mechanisms, communication infrastructures, and user-side and services-side identity management (PRIME 2008). Some of the concepts were incorporated in prototypes, others are to be considered in follow-up projects the PRIME partners are involved in.

29.3.2.2 Personalization

Mary enjoys the ease and efficiency with which she can log on to her personalized view of the medical file application to make appointments with health care providers and review her medication information and medical summaries. Personalizing interaction involves the use of information about the user to alter the content presented to provide value to the user (Karat, Karat, and Brodie 2004). In their empirical studies, Karat et al. found that the most critical element in the willingness of users to adopt and use personalization systems was user control of personal data. Users want to be in control of their privacy and appreciate the benefits of personalization when they have control over the use of their personal data. A case study of the personalization research (Karat and Karat 2010a,b,c) illustrates the central role that explicitly stated privacy policies have in the success of personalization in an e-commerce web application. End users wanted to know that they had control of all data kept in their profile and the ability to review and edit it at any time. Also, end users strongly endorsed a privacy and personalization policy to collect only the minimum amount of PI necessary for the application to provide the valued personalization functionality to the end user. Once end-user concerns regarding privacy were satisfied, then additional variables related to the particular personalization feature available, user characteristics, and the business context of a task influenced the value of personalization to the user.

Cranor (2004) complements this line of research with an analysis of privacy risks associated with personalization and presents a set of design guidelines to reduce privacy risks in personalization systems. Teltzrow and Kobsa (2004) completed a meta-analysis of 30 consumer studies of user privacy preferences regarding personalization systems and found that consumer demands and current practice diverge significantly regarding control of PI. The vast majority of businesses neither allow control over what information is stored or ability to access it for verification, correction, or updates.

29.3.2.3 Anonymity

Anonymity and anonymization in IT includes the ability of a user to maintain privacy while completing transactions on a network and the ability for a user to keep the data they provide from identifying them personally. In the medical scenario, Mary is willing to provide her medical data to the national database conducting research on the disease she has, given that it is provided in an anonymized manner.

Sweeney (2002) and Malin and Sweeney (2004) have demonstrated that minimal amounts of information believed to be anonymous can be used to personally identify an individual. Since data are aggregated from a myriad of databases in the networked world in which we live, users may falsely believe that they can remain anonymous by providing only minimal bits of information in transactions here and there over time. When the data are aggregated though, the person can be identified fairly easily based on three minimal pieces of information. Users may not think about the types of data from different sources that might be combined and analyzed to identify them, and thus there is a false sense of privacy. This research also makes it clear that there is a grey area in terms of what data elements might be labeled as PI versus personally identifying information—it depends on the other data available in context.

In researching anonymizing networks, Dingledine and Mathewson (2005) conclude that in order for users to be able to preserve their privacy on networks by completing transactions without revealing communication partners, it is critical that the system chosen is usable so that other users can successfully use the system as well. Anonymizing networks are successful at hiding users among other users. An eavesdropper may be able to determine who is using the network, but cannot identify who completes a particular transaction. Moreover, the larger the group of users on the network, the more anonymous the participants become. The catch phrase in this area is "anonymity loves company" (Reiter and Rubin 1998, p. 70).

29.3.3 Organizational Requirements in Managing Privacy

The topic of organizational requirements in managing privacy includes several related subtopics as follows:

1. The three pillars of people, technology, and business processes
2. The ongoing training and education of personnel
3. Policy technology as an essential component of privacy solutions
4. Ongoing business process refinement in a new world of information sharing
5. Compliance audits: Monitoring of privacy policies in operation

29.3.3.1 Three Pillars of People, Technology, and Business Processes

The successful implementation of a privacy program in an organization is supported by people, technology, and business processes (Karat and Karat 2010a,b,c). A privacy program cannot succeed in a modern day organization with only one of these pillars, and will fail if any one of these foundations is compromised. All three are required and they are the minimum resources necessary to handle an organization's privacy program. In the medical scenario, Mary learns about and trusts the privacy policies used by the health care group practice within their group, and the policies about sharing her data with other medical specialists involved with her care, the insurance organizations handling her claims, and the national health care center studying her disease. She feels a level of comfort in knowing about the policies, understands that the MedicalFile application enforces the policies through technology, and is pleased that the group practice monitors and audits the policies in practice.

29.3.3.2 Ongoing Training and Education of Personnel

The employees or members of an organization must have the opportunity to learn the privacy policies of the organization regarding the use of all data that people in the organization collect, use, manage, store, access, view, share, disseminate, and destroy. The workings of an organization fundamentally rest with the people in the organization; they are the heart of it. All data handling and processes do not happen automatically, and organizations can only go so far in controlling the data that employees see as part of manual and automated business processes. The people are the first and last line of defense in the organization. If they understand the intent of the organization's privacy policies, they can go above and beyond to ensure that the intent of the policies is executed in day-to-day transactions in the organization. Also, all the people who interact with the organization must have an easy and efficient means of learning about the organization's privacy policies and the implications of the privacy preferences they select in interactions with the organization. The training and information available to people within and outside the organization is an ongoing process that requires updates as external and internal requirements and situations change.

29.3.3.3 Policy Technology as an Essential Component of Privacy Solutions

As mentioned previously, the adoption of technology has brought mankind closer to the vision of a ubiquitous computing world (Weiser 1991). As data are collected, stored, used, disseminated, and destroyed around the world, it is essential that end users and organizations define and agree to the policies for the handling of the data and that these policies conform to legislation at all levels of government in the geographies in which data are involved. Organizational privacy policies, particularly for governments and multinational corporations handling data around the world, can be complex. For particular pieces of legislation in the United States, for example, a privacy policy may consist of thousands of policy rules. Therefore, there is a pressing need for usable policy technology that enables organizational users to author policies, analyze, deploy, and audit privacy policies in operation.

The privacy technology must be architected and designed with these requirements in mind, so that it is easy for the intended users to work with, integrates with an organization's business processes, and can be modified and audited as desired is essential (Karat et al. 2005).

29.3.3.4 Ongoing Business Process Refinement in a New World of Information Sharing

Privacy policy technology must be able to keep pace with the fast pace of change in organizations today and in the future. For example, external events, new legislation, or inventions within the organization may significantly change the products and services that an organization provides, and these changes will necessitate updates and modifications to the privacy policies deployed by the organization. As the organization's mission changes, the business processes and applicable privacy policies must be updated as well.

29.3.3.5 Compliance Audits: Monitoring of Privacy Policies in Operation

For the ongoing management of the organization, there must be the ability to complete compliance audits of the operation of the privacy policies in the day-to-day execution of the organization's business (Karat et al. 2005). Compliance audits of the enforcement of the privacy policies in the organization's operations will enable the organization to grow and learn about gaps in policies and emerging changes requiring new business processes. The basic purpose of the audit is to determine whether the privacy policies are working as intended in operation. Audits can uncover exception processing that may be the result of gaps in policies, internal fraud and abuse, or changes in the external environment that necessitate changes in business processes and the associated privacy policies for them. Compliance audits of policies close the loop in the policy lifecycle from policy definition to compliance audits.

HCI research focused on the organizational view of privacy includes the Server Privacy Architecture and Capability Enablement (SPARCLE) Policy Workbench (Karat et al. 2005; Karat and Karat 2010a,b,c), privacy policy research across the domains of health care, finance, and government (Brodie et al. 2008), policy refinement research in the Open Collaboration Research project on privacy and security policies (Bertino et al. 2009; Karat et al. 2009; Ni et al. 2008), and privacy research in the health care domain by Adams and Blanford (2005). Karat et al. (2005) used user-centered design methods with 109 target users to identify organizational privacy requirements, and then designed and tested a prototype system for users to author privacy policies, implement the policies, and conduct compliance audits of them. The prototype was a Wizard-of-Oz version of a system, meaning that users were able to have an immersive and dynamic experience with the system capability that seemed real, although there was no functional code behind the screens. Empirical results show that organizational users value highly a set of capabilities that enables policy officers to use natural language to author the policies, with the option of beginning with a template. Users found a visualization of the policies very valuable for communication, review, and identifying necessary modifications to rules, and reaching consensus on policies across the organization. The implementation capability enabled the experts to approve the nominated mappings between rule elements and data base fields and applications in the organization's configuration. The compliance audit capability enabled the users to run general audits to verify that the policy was complying with regulations and was being enforced operationally as stated in the policy, and to run specific inquiries based on individual requests for information about use of PI by the organization. The organizational users who were the participants in the evaluation study rated the prototyped functionality as being of very high value to them.

Karat et al. (2006) conducted an empirical study to determine whether the two methods of authoring rules that were prototyped by Karat et al. (2005) (natural language with a guide or structured list) enabled users to create higher quality rules than an unguided natural language control method (using a word processing window). Empirical results demonstrated that both prototyped authoring methods yielded higher quality rules than the control condition. Users, with no training in use of the methods, were able to create privacy rules covering about 80% of the required rule elements described in scenarios when using either the natural language with a privacy rule guide tool or a structured list tool compared with covering about 40% of the required elements in scenarios using the unguided natural language control condition.

Research and development on the SPARCLE Policy Workbench continued and the general rule authoring utility for the privacy domain became functional, with working code that was successfully tested with the privacy policies of banking/finance, health care, and government organizations (Brodie, Karat, and Karat 2006; Karat, Brodie, and Karat 2005). Users could create new policies with SPARCLE, import existing text versions of policies, or cut and paste sections of policies to form a new policy. Then SPARCLE used natural language processing technology to parse and identify the rule elements. Users reviewed and modified the rules, and when they were happy with the policy, the policy was transformed into XACML code (the OASIS international standard for the format of security access control rules with a privacy profile) for input to the enforcement engine. The team continued to design and create other components in the end-to-end solution for privacy and believed that the policy workbench could be generalized to the security domain and possibly others. The functional SPARCLE prototype was transferred into development and the first SPARCLE-based product called Secure Perspective was available commercially in 2007 (IBM Secure Perspective 2007). The initial product focused on security policies due to market and other considerations. Other product releases supporting various platforms and functionality were released in 2007 and 2008, respectively.

Another policy research effort focused on both the privacy and security domains and involved collaboration between IBM Research, Carnegie Mellon University, and Purdue University (Bertino et al. 2009; Karat et al. 2009; Ni et al. 2008). In this research, the combined team worked on the architectural framework for privacy and security policies and investigated issues in the refinement of policy from high-level statement in natural language to the policy execution layer. Once policies are defined, they need to be transformed to tie the intent of the policy to the system objects and run in real time (Karat et al. 2009). This transformation will move a policy rule from a natural language statement that is understandable by people such as the following:

> Healthcare staff can forward patient medical information for the purpose of national medical research if the information is anonymized.

to a rule format for processing by the organization's computer systems:

```
If request(upload(DBname)) && BName==NIH_Records
then Set DB = anonymize(PatientDB,NameAttribute,
AddressAttribute); upload(DB,DBname)
```

Policy rules may be written using a variety of formats. A new format being researched is privacy-aware role based on access control (Ni et al. 2008). After policies are authored and prior to the privacy policy rule enforcement system going live, a set of policy rules can be analyzed to determine any conflicts or redundancies with other privacy policies already deployed or part of the new set being created (Bertino et al. 2009). Research investigated HCI design ideas related to system feedback to the user on how to resolve identified policy conflicts and redundancies in policy sets.

Adams and Blanford (2005) addressed the gap between organizational and user perspectives of security and privacy in the healthcare domain. They conducted ethnographic evaluations of 93 clinical staff, managers, library staff, and IT department members in two hospitals. They found contrasting perspectives on security and privacy within the groups. They use the concept of "communities of practice" to identify security and privacy issues within organizations, acknowledging the importance of knowledge gained through a work community in day-to-day work practices and formal learning. The "community of practice" acts as a link between the individual and the organization. In their research, one hospital was able to improve communication and awareness across the organization through a community and user-centered approach to the design and development of an organizational privacy and security application. In the second hospital, a clash between the formal rules and the perspective of the local community of practice were at the heart of an identified security problem. The authors highlight the importance of designing security and privacy practices within the context of communities of practice.

TABLE 29.1
Design Factors in Usable Privacy Solutions

Networked World

- There is technology and data collection in most aspects of everyday life in the developed world.

Privacy Legislation

- Requirements for privacy vary by geography and across cultures. Organizations and end users must become knowledgeable.

End-User Views of Privacy

- People perceive and define privacy intrusions differently, and privacy solutions must be designed to support varying definitions and choices.
- End users must be informed of the risks of choices involved in different anonymity enabling situations.
- User control of data to be disclosed is critical for using personalization.
- Regarding pervasive devices, users want control over access to them but want it to be easy to use, fast, and flexible.
- Regarding social computing, user must be clearly informed about data at risk of disclosure.
- For the end user, managing privacy policies is not generally his or her main task; it is a necessary step in completing tasks or activities of value to the user. Managing privacy needs to be simple and efficient.

Organizational Management of Privacy—Requirements, Approaches, and Perspectives

- Authors of privacy policies, who have the knowledge of organizational practices, must be able to tie written policies to implementation through technology, with verification through compliance audits.
- Need views into policy applications for employees within organizations, and for end users who interact with them.
- Automated transformation of policies from natural language to executable code is desirable, with clear links between the transformation levels.
- Policy refinement must include ability to complete policy analysis on current and future policies. The end user and organizational users would benefit from information about possibilities for resolving conflicts and redundancies in privacy policies.
- Need to be able to demonstrate the impact of a policy and ensure that implementation will execute the policy intent of the organization.
- Policy architecture must allow for simple and fast updates of organizational privacy policies based on changes in end-user preferences, legislation, external events, and changing business situations.
- The intent of privacy policies needs to be understandable across cultures.
- There is a critical need to understand community of practice issues in an organization in creating privacy policies.

29.3.4 Summary of Human–Computer Interaction Privacy Research Results and Recommendations

A body of HCI research that can inform the design of usable privacy mechanisms is growing (see Table 29.1). This review of research has covered topics related to the networked world in which we live, the end-user's view of privacy and concern for anonymity while interacting over networks, desire for personalization, and the impact of social networking technologies on privacy. The research on organizational views of privacy has investigated how to provide people technology to

- Create privacy policies that people can understand
- Enable an organization to implement the privacy policies in their computer systems

- Provide simple and efficient ways to update policies as changes occur
- Enable auditing of privacy policy enforcement events for compliance with organizational and legislative requirements

In our roles as usability professionals, it is incumbent on us to consider the issues of privacy in the design, review, approval, development, and use of computing systems by end users and organizations.

The HCI field will be enriched by considering privacy more fully, and this research will cross-pollinate in several areas. For example, cross-cultural studies of privacy can inform the growing interest in HCI in cross-cultural interfaces and coordination. Kumaraguru and Cranor (2005) report a study that demonstrates an overall lack of awareness of privacy issues and less concern about privacy in India than has been found in similar studies conducted in the United States.

29.4 VIEWS ON SECURITY

It is broadly recognized that one of the major challenges to the effective deployment of information security systems is getting people to use them correctly. As far back as the 1970s, usability with specific reference to security mechanisms was identified as a key principle in the design of secure systems (Saltzer and Schroeder 1975). Even beyond the domain of electronic information systems, there are many examples of the fact that overly complex security systems actually reduce effective security. For example, Kahn (1967), cited by Anderson (1994), suggests that Russian military disasters of the Second World War were partly due to the fact that Russian soldiers abandoned the official army cipher systems because they were too hard to use, and instead reverted to simpler systems that proved easier to crack. Ellison and Scheiner (2000, p. 4) sums up the situation: "Security measures that aren't understood by and agreed to by everyone don't work." However, as with many areas in the design of complex systems, recognizing that there is a potential problem does not necessarily create a rush of work to resolve it. Work on making security usable—and balancing the complex relationship between "secure" and "easy to use" is just beginning to become a major topic.

Networked computer systems are increasingly the site of people's work and activity. Millions of ordinary citizens conduct commercial transactions over the Internet, or manage their finances and pay their bills online. Companies increasingly use the Internet to connect different offices, or form virtual teams to tackle mission-critical problems through entirely "virtual" interaction. For example, interaction between citizens and local and federal government agencies can increasingly be conducted electronically; and the 2004 national elections in Brazil and (to a much more limited extent) the United States saw the introduction of electronic voting, which will no doubt become more widespread.

However, these new opportunities have costs associated with them. Commercial, political, and financial transactions involve disclosing sensitive information. The media regularly carry stories about hackers breaking into commercial servers, credit card fraud, and identity theft. Many people are nervous about committing PI to electronic information infrastructures. Even though modern PCs are powerful enough to offer strong cryptographic guarantees and high levels of security, these concerns remain.

The need for secure systems is broadly recognized, but most discussions of the "problem of security" focus on the foundational elements of information systems (such as network transmission and information storage) and the mechanisms available to system developers, integrators, and managers to ensure secure operation and management of data. Security, though, is a broader concern and a problem for the end users of information systems as much as for their administrators. Participation in activities such as electronic commerce requires that people be able to trust the infrastructures that will deliver these services to them.

This is not quite the same as saying that we need more secure infrastructures. De Paula et al. (2005) suggest that it is important to separate theoretical security (the level of secure communication and computation that is technically feasible) from effective security (the level of security that can practically be achieved in everyday settings). Levels of effective security are almost always lower than those of theoretical security. A number of reasons for this disparity have been identified, including poor implementations of key security algorithms insecure programming techniques (Wagner et al. 2000; Shankar et al. 2001), insecure protocol design (Schneier and Mudge 1998), and inadequate operating systems support (Bernaschi et al. 2000).

One important source of the disparity, though, is problems around the extent to which users can comprehend and make effective use of security mechanisms. Approaches that attempt to make the provision of system security "automatic" or "transparent" essentially remove security from the domain of the end user. However, in situations where only the end user can determine the appropriate use of information or the necessary levels of security, then this explicit disempowerment becomes problematic.

Perhaps the best consolidation of usability issues for security comes from Sasse and Flechais (2005). They observe that currently users often disclose (or write down) passwords, fail to encrypt confidential messages, switch virus checkers off, and generally engage in behavior counter to "good security," and ask why this is so common. They conclude that most users

1. Have problems using security tools correctly
2. Do not understand the importance of data, software, and systems for their organizations
3. Do not believe that the assets are at risk
4. Do not understand that their behavior puts assets at risk

Whitten and Tygar (1999) identify a "weakest link property," stating that attackers need only to exploit a single weak point in system security. Frequently, the human user proves to be this weakest link—not from malicious intention but from inability to reasonably do the right thing. Sometimes, this is a matter of education—people need to be made aware of what to do and why to do it. But some of the blame and burden on making our systems more secure rests in making security systems more usable.

29.4.1 Usability of Security Software and Mechanisms

In a series of studies, researchers at University College, London, have explored some of the interactions between usability and security (Adams and Sasse 1999). They focused on user-visible elements of security systems, such as passwords. Although many information systems professionals regard users as being uninterested in the security of their systems (and, indeed, likely to circumvent it by choosing poor passwords, etc.), Adams and Sasse's investigations demonstrate that users are certainly motivated to support the security of the system, but often unable to determine the security implications of their actions. The specific problems that they identify with passwords have also led to interesting design alternatives (Brostoff and Sasse 2000; Dhamija and Perrig 2000).

In some cases, the complexity of making security work is as much a matter of interface design as anything else. Whitten and Tygar (1999) present a usability analysis of PGP 5.0, demonstrating the difficulties that users have in completing experimental tasks (in their user study, only 3 out of 12 test subjects successfully completed a standard set of tasks using PGP to encrypt and decrypt e-mail.) The problems that they uncovered were largely problems of interface design, and in particular, the poor matching between user needs and the structure of the encryption technology provided to meet these needs.

Zurko and Simon (1996) explore similar concerns in their focus on "user-centered security." Their work addressed their perception that the inscrutability of conventional security mechanisms makes it less likely that users will use them effectively. The approach they outline focuses on graphical interfaces and query mechanisms to MAP, an authorization engine. Although this approach is clearly helpful, it is limited to a particular area of system security, and lacks the real-time feedback.

A classic research paper by Karat (1989) demonstrates the valuable contribution that HCI can make in the design of security solutions. Karat, as the HCI lead on a security project, worked in collaboration with the security technical staff to create a security application to be used by branch office personnel in IBM. She joined the team that was already underway on the creation of a new security application whose goal was to eliminate the recurring need for security authorization while performing discrete but related tasks that composed a business process for the employees. The current system impacted user productivity, and due to human memory load issues in its design, added a level of risk regarding the security of the application, and data that needed to be resolved. Dr. Karat conducted a usability evaluation of the initial design of the solution by the security technical staff. Results showed that only 20% of the target users could successfully sign on to the system, and those users required over 3 minutes to do so. She identified four key usability problems in the design of the system. Working with the security technical staff and developers, she was able to improve the design of the user interface to the security application while the security staff identified a technical solution to finesse a key complexity issue in the underlying system. Together the team created a successful solution. Usability tests of the final design showed that 100% of the target users were able to sign on successfully and begin the selected transaction on application within 7 seconds. The new application was deployed on time and under budget because the usability problems were discovered early and resolved. There was high user satisfaction with the security application, and the need for a help desk in the transition period was eliminated. The cost-benefit analysis on the use of HCI skills on the project demonstrated a 1:10 ratio; for each $1 dollar spent on usability, $10 in costs were eliminated from project development and application deployment (see Karat 1994, 2005 for discussion of usability cost-benefit methodology and data).

29.4.2 User Control over Security

How might Mary see her role in the protection of the systems that collect and house her medical information? At one level, Mary wants what we all want. She wants to know that the systems are "secure"—that they have safeguards against misuse, that those she trusts protect those systems, and that she does not have to do much to ensure the security. If Mary asks for a copy of her information, she expects that the system will recognize her and grant the request. If someone Mary does not know asks for her information, she would expect that the system would not grant the request without checking with her. She might expect that if her doctor's laptop computer was lost or stolen, her information would not be easily retrieved from it. In the language of systems security, these can be seen as issues of access control and data encryption. In some cases, we try to remove the need for humans to ensure system security, in others they are a necessary component.

One area at the intersection of usability and security that has received some attention is the role of access control in interactive and collaborative systems. For example, Dewan and Shen (Dewan and Shen 1998) have explored the use of access control and meta-access control models as a basis for describing and controlling degrees of information access and management in collaborative systems. This is not simply a technical matter since the structure and behavior of these "internal" components can have a significant effect on the forms of interactivity and collaboration they can support (Greenberg and Marwood 1994). As with privacy, policies play an important role in security through access control. There is no single answer to how access should be controlled in a system. Context determines the access control that is appropriate and this control is communicated through human-authored policies.

Many collaborative systems involve privacy issues and need to provide users with control over the disclosure of information. This has spurred a number of researchers to explore the development of privacy control systems that are tailored to the needs of end users. For instance, Dourish (1993) describes the relationship between three different security mechanisms for similar multimedia communication systems, each of which reflects assumptions and requirements of the different organizations in which they were developed. Bellotti and Sellen (1993) draw on experiences with multimedia and ubiquitous computing environments to identify the source of a number of potential privacy and security problems. Their primary concepts—disembodiment and dissociation—are both visibility problems, related to the disconnection between actors and actions that renders either actor invisible at the site of action, or actions invisible to the actor.

Based on their investigations of privacy problems in online transactions, Ackerman and colleagues propose the idea of privacy critics—semiautonomous agents that monitor online action and can inform users about potential privacy threats and available countermeasures (Ackerman, Cranor, and Reagle 1999). Again, this mechanism turns on the ability to render invisible threats visible.

One important related topic is control over the degree of security available. One of our criticisms of traditional security systems has been their “all or nothing” approach. However, there has been some work that attempts to characterize degrees of security provision, as embodied by the idea of “quality of security service” (Irvine and Levin 2001). This builds on earlier work establishing a taxonomy of security service levels (Irvine and Levin 1999). The fundamental insight is that organizations and applications need to trade-off different factors against each other, including security of various forms and degrees, to make effective use of available resources (Thomsen and Denz 1997; Henning 1999). Although this work is directed toward resource management rather than user control, it begins to unpack the “security” black box and characterizes degrees and qualities of security.

For end users, perceived security can be defined as the level of security that users feel while they are shopping on e-commerce sites. Yenisey, Ozok, and Salvendy (2005) report a study that aimed to determine items that positively influence this feeling of security by users during shopping, and to develop guidelines for perceived security in e-commerce. An experiment allowed users with different security assurances to shop on simulated e-commerce sites. The participants were divided into three groups, shopping for cheap, mid-range, and expensive products, respectively. Following the shopping environment, a virtual shopping security questionnaire was presented to the users. Generally, there were no significant differences in item ratings between the groups of different shopping item values. A factor analysis procedure determined two main factors concerning perceived security in e-commerce. The perceived operational factor includes: the site’s blocking of unauthorized access; emphasis on login name and password authentication; funding and budget spent on security; monitoring of user compliance with security procedures; integration of state-of-the-art systems; distribution of security items within the site; website’s encryption strategy; and consolidation with network security vendors. The perceived policy-related factor includes: the website’s emphasis on network security; top management commitment; effort to make users aware of security procedures; the website’s keeping up-to-date with product standards; the website’s emphasis on security in file transfers; and issues concerning the web browser.

29.4.3 Research Topics for Human–Computer Interaction in Information System Security

The computer science research community commonly views security and privacy as related topics (e.g., the IEEE Symposium on Security and Privacy is a major research forum). In this literature, privacy gets much less attention that the wide range of security topics. This can be seen as related to an effort to make system secure against human error (or attack) as much as possible. There are some areas of secure systems research that have given more attention to HCI issues. We summarize some of these in Section 29.4.3.1.

29.4.3.1 Secure Design and Administration

Yee (2002) provides guidance for designing and evaluating usable secure software aimed at protecting the interests of the legitimate user. Starting with the work of Saltzer and Schroeder (1975), he divides his guidelines into two general categories—guidelines for authorizing others to access valuable resources and guidelines for communication between the system and the user. For authorization, he suggests associating greater risk to the user with greater effort or less visible operations so that user’s natural tendencies leads to safe operations. For communication, he suggests enabling users to express safe security policies that fit their tasks. The guidelines are not empirically derived, but are gleaned from the experiences of security software designers.

Kandogan and Haber (2005) have applied ethnographic approaches to the study of a particular class of users involved in the protection of systems—security administrators. The issues are somewhat different for populations whose job responsibility includes the security of systems used by others when compared with the situation in which users are acting as their own “security administrators.” However, they represent another important category of user in the area of creating usable security systems. In the ethnographic tradition, they examine their users in typical work situations (e.g., detecting and addressing a security attack) and look at how well their tools help them to do their job. One important result of their work is the realization that administrators simply have too much to look at in the course of their daily work, and that they specifically need tools that help them understand vast amounts of dynamic data. Their work focuses on developing visualization tools to assist in this situation.

Balfanz et al. (2004b) present guidelines for usable security systems developed over a course of research that has focused on the intersection of security and usability. The guidelines include advice to focus on designing both security and usability into the system rather than attempting to retrofit either into existing systems. It seems that the lessons learned in HCI work with regard to the need to consider it early and often during design also apply to security (security engineering is an important topic in that field, much as usability engineering is in HCI). Additional guidelines also call for a focus on the user by looking for high-level building blocks that can be used to create user-oriented solutions, rather than assuming users will be skilled at assembling their own tools for effective security. They point out that the security community has long held the belief that security is more important than user needs, and that it is users who must adapt to requirements to assure system security. An empirical study (Balfanz et al. 2004a) showed how hard it can be for security researchers to accept the difficulty of using some of their own systems. When they evaluated the time it took to secure a wireless network using a tool they had developed, they were surprised to find that it took users over 2 hours on average. The authors mention that the reaction in the security community was often that the "empirical data must be wrong." Although these views are changing with the emergence of usable security research groups like this one, we should not expect the difficult trade-offs to be quickly resolved.

Maxion and Reeder (2005) provide a study in which a system specifically designed to assist users in avoiding errors (Salmon) is compared with the standard operating system interface for making file-sharing decisions. Salmon was found to increase successful task completion by 300% (the standard interface used misleading terminology that caused users to feel that they were sharing files when in fact they were not). Users also spent less time searching for information using Salmon and had a greater proportion of their time on essential task steps rather than security overhead. In their study, they demonstrate that attending to error avoidance in interface design can facilitate usable security.

29.4.3.2 Authentication Mechanisms

Security systems are designed to let authorized people in (the permission problem), and to keep unauthorized people out (the prevention problem). This involves three distinct steps: (1) identification, (2) authentication, and (3) authorization (Renaud 2005). The identification step asks a person to identify himself—usually by means of a token or an identification string such as an e-mail address or account number. Once the identification token has been provided, the person has to provide some evidence of his identity (authentication). This can be done by presenting something they know (e.g., password), something they recognize (e.g., graphical passwords), something they hold (e.g., a certificate), or something they are (e.g., biometrics). For the first three, authentication depends on the user and the system sharing a secret, which for security purposes should be difficult for someone else to guess, and for usability purposes should be easy for the user to remember. The trade-off between these two purposes can be difficult to resolve, and contributes to discussions of "how much security" rather than a view of security as something that is either present or not. In the case of biometrics, the system records a digital representation of some aspect of a person's physiology or behavior at enrollment, and this is confirmed at authentication time.

Many authentication scenarios can be strengthened (in a security sense) through the use of public key cryptography. For example, a user can have a smart card that contains a public key and a matching private key. Instead of a password, the user's public key can be placed on file at a remote computer system (authentication server). To authenticate the user, the remote system sends the user a random challenge. The user "signs" the challenge with his private key and sends the result back to the remote server, which verifies the signature with the public key. In this way, the remote server can verify that the user has possession of the private key without having to receive it. Instead of having the public key on file at the remote system, the smart card can submit both the signed challenge and a public key certificate that has been signed by a third party. In this case, the use of public key technology is called public key infrastructure.

Whatever methods are used, at each stage of an authentication process, we can ask "is it secure?" The real areas of vulnerability are the input mechanism and the user. In the case of knowledge-based authentication, the user must be able to keep the secret and the secret must be hard to discover.

29.4.3.3 Passwords

Random passwords are currently the most popular user authentication mechanism. As such, they also represent the most common research target for work in the security and usability arena. One might consider them the "white rat" for HCI work, similar to menu structures or word processor designs of earlier HCI eras. Yan et al. (2005) conducted research to examine some of the commonly held beliefs about the security and usability properties of various password generation guidelines. In their work, they confirmed that passwords generated by typical advice (e.g., use of a random sequence of letters and numbers) can be difficult for people to remember, but that more cognitively friendly advice (such as select mnemonic-based passwords) can be much easier to remember without sacrificing the theoretical security levels of random passwords. They also explored additional security screening capabilities—such as screening for "weak choices (e.g., obvious dates/places)," and found that such filtering could work well with other advice provided to users. Contrary, the beliefs held within security communities, they did not find that theoretical analysis necessarily corresponded to actual system security. For example, encouraging the use of truly random passwords resulted in more frequent writing down and carrying passwords than encouraging the use of mnemonic passwords.

Others have taken up the challenge of providing usable passwords through nontext password schemes. For example, Monrose and Reiter (2005) provide an analysis of graphical passwords. As with text passwords, graphical passwords might be selected by the user (possibly enhancing usability but perhaps decreasing security) or selected by the system (enhancing security at the cost of usability). Although there is certainly ample evidence that people exhibit powerful memory for images (e.g., Mandler 1991), it is not completely clear that this translates easily into a superior password mechanism. Results from Monrose and Reiter demonstrate that graphical password schemes can suffer from drawbacks similar to those of textual schemes. People tend to select memorable graphical passwords enabling them to be more easily attacked, and random graphical figures can be more secure but also more difficult to recall.

Wiedenbeck et al. (2005) have experimented with an interesting variant of graphical password. In their PassPoint system, users are provided with an image in which they establish a password by selecting a series of points on the image (the number of points selected and the order in which they are selected are factors in the complexity—and thus the potential security features of the password). Their experimental work indicates that such a system is promising from a usability perspective—people can learn the systems easily and remember their sequences over time. However, performance with such a system was not as good as with a comparison textual password system. It took more time to learn the passwords, and more time to enter them even with practice. The extent to which such performances differences might be due to novelty associated with graphical systems compared with textual systems remains unclear, and is in need of further research before we might expect them to replace the textual passwords we have so much experience with. As De Angeli et al. (2005) report in their work on the graphical authentication systems, successful design of password systems is a complex task and requires considering and weighing a number of factors (such as the trade-off between security and usability).

Recently, Everitt et al. (2009) tackled some of these broader graphical password issues in the first study of multiple graphical passwords to systematically examine the effect of frequency of access to a graphical password, the effects of interference resulting from interleaving access to multiple graphical passwords, and the effect of patterns of access while training multiple graphical passwords.

In this work, Everitt et al. show that field studies of graphical password systems are likely to overestimate ease of access if they do not study the realistic use of multiple graphical passwords. With regard to interference, participants in their study who accessed four different infrequent passwords each week had a failure rate more than 10 times greater than participants accessing a single infrequent password. Everitt et al. only used facial recognition, and it is possible that other schemes (e.g., other location or graphic-based schemes) might be less susceptible to interference. Still, developers of graphical password systems should consider the guidance to study ease of access under more realistic training and use conditions.

Inglesand and Sasse (2010) have also answered this call to more realistic evaluation of password policies and practices. Their work presents a study that re-examined password policies and password practice in the workplace. Staff members in two organizations kept a password diary for 1 week, which produced a sample of 196 passwords. The diary was followed by an interview that covered details of each password, in its context of use. The authors found that users are in general concerned to maintain security, but that existing security policies are too inflexible to match their capabilities, and the tasks and contexts in which they operate. As a result, these password policies can place demands on users that impact negatively on their productivity and, ultimately, that of the organization. Inglesant and Sasse (2010) conclude that, rather than focusing password policies on maximizing password strength and enforcing frequency alone, policies should be designed using HCI principles to help the user to set an appropriately strong password in a specific context of use.

Some of the trade-offs between security and usability, and some of the tension between the approaches of the two communities can be seen in some recent research lines. Davis, Monrose, and Reiter (2004) provide a study that looks at how user selection of passwords in graphical password schemes might increase the likelihood that passwords can be attacked. For text passwords, user selection is the norm, and there is no evidence that allowing users to select passwords (following some guidelines for length and character type) makes the resulting passwords any more vulnerable to brute force attack than the theoretical maximum set of passwords for the set of characters involved. Several new approaches for password selection—including a number that involve allowing the user to select a sequence of images—have been developed to help with password memorability (see Real User Corporation [2002] for a commercially available example). Although there has been research on several of these approaches to indicate that they provide passwords that are more memorable than text passwords, and have similar theoretical protection (entropy), there has not been much research into how effective they are in actual use. The research reported by Davis et al. suggests that people do not select as randomly, and instead form passwords with a limited set of possibilities (perhaps analogous to forming passwords with strings that might be easily associated with an individual such as birthdates). Overall, many questions remain regarding how to optimize both security and usability at the same time in the area of knowledge-based passwords.

29.4.3.4 Biometrics

As we have discussed, authentication in computer systems has suffered from the limited number of types of mechanisms available for a system to know who it is interacting with. With advances in technology, we have new opportunities (beyond traditional mechanisms such as passwords) for systems to recognize us. Biometrics refers to a means of identification that can be uniquely associated with an individual (e.g., voice patterns, fingerprints, and hand geometry). Biometrics involves the comparison of live anatomical, physiological, or behavioral characteristics to a stored template of a person (Coventry 2005). Physiological biometrics that have been investigated include those based on fingerprints, hand and/or finger geometry, and the patterns of retinas, veins, irises, and faces. Behavioral biometric techniques that have been advanced include those based on voice, signature, and typing behavior (Peacock, Ke, and Wilkerson 2005). Other techniques that are not yet as well developed include recognition of characteristics such as ear shape, gait, laughter recognition, facial thermograms, and lip shapes.

To begin, it is important to distinguish between the use of biometrics for identification and verification. For identification, the task is to identify an individual out of a population of all possible users. As the population of possible users grows, the demands on identification can also grow (either in terms of performance or accuracy of the recognition system—challenges similar to speech recognition for large vocabulary unconstrained speech). For verification, the task is to verify a particular identity by matching a characteristic to a stored template for the individual. The computational task is much simpler for verification than for identification (where only fingerprints, retinal scanning, and iris scanning have been proven successful for large populations). For the most part, we will talk about the use of biometrics for verification in this chapter.

Coventry (2005) provides an excellent illustration of how biometrics might be used and what some of the current trade-offs are for a common application—ATMs. For all techniques, there are usability issues that can be cited as barriers to implementation and acceptance. Fingerprint identification is relatively well developed, but sensors placed in public places are subject to environmental issues (such as dirt or latent prints) and user training issues (they can be sensitive to where the finger is placed on the reading device). Retina scans have attractive potential, but are currently quite invasive and difficult to develop for a full range of users (they require placing the eye close to a sensor location). Speaker verification can be subject to background noise problems, and relatively complicate user registration requiring more input than a single word or phrase. Signature verification is attractive because of its long use and association with financial authorization, but current technology is not yet seen as reliable. Typing verification is seen as having some potential, but lacking the reliability necessary for large population verification.

Biometrics researchers have determined that real-life users are the biggest variable in system performance. Ashbourn (2000) has suggested several user characteristics for evaluating biometric systems. These include a users general acceptance of the biometrics concept used (i.e., Is the idea of having a system read such characteristics acceptable?), general knowledge of the technology (i.e., Does the user understand what is being done?), knowledge of the particular biometric characteristic (e.g., Does the user understand that if fingerprints are being read, the finger should be centrally placed on the sensing device?), experience with the sensing device, environment of use (e.g., public or private), and transaction criticality. Although biometrics technologies are rapidly improving, inherent performance limitations remain and are extremely difficult to work around, except perhaps by combining multiple technologies or providing for a bypass.

29.4.3.5 Other Security Approaches

One other way of authentication by "something I have" is through the use of smart cards—here referring to a portable device (or card) with processing power and authentication information contained on it. Smart cards essentially add an integrated circuit to familiar plastic credit cards enabling the use of cryptographic services like random number generation and public key cryptography (Piazzalunga, Salvaneschi, and Coffetti 2005).

Just (2004) examines the design of challenge-question systems, looking for ways to improve the usability/security characteristics of these. Beckles, Welch, and Basney (2005) look at the usability of security mechanisms in grid computing contexts.

Encryption of data is a well-established mechanism for protecting information. Rather than storing or transmitting data that can be read "in the clear" (i.e., data that is stored in standard code schemes such as ASCII), information is encrypted using a key, and then must be decrypted using a key. Although this contributes to achieving some security goals (e.g., if an encrypted file containing sensitive information is lost or stolen, it would be a considerably harder task to make use of the information than if a plain data file were lost or stolen). There is a "usability cost" for this added security—users might have to indicate when and how files are to be encrypted or decrypted. Unfortunately, the complexity of this process has been found to cause user difficulties, variously resulting in inability to use security mechanisms appropriately (Whitten and Tygar 1999; Caloyannides 2004; Guttmann 2003). Work continues at addressing the issues associated with cryptographic approaches to security (e.g., Balfanz, Durfee, and Smetters 2005).

29.4.3.6 Security Policy Management

Much of the research related to policy management mentioned above in the Privacy Research Section also applies to Security Policy Management. To the extent that privacy policies can be viewed as a specific case of an access control policy, issues in authoring, conflict resolution, visualization, and other issues clearly overlap. There are additional complexities for privacy policies in that they include an additional element (purpose of use of the resource in question). These complexities drive implementation challenges that, to date, have not been fully addressed. For example, how would a system know that Mary's doctor was using her medical information for authorized purposes and not for unauthorized use? At some level, this is a concern for security policies as well. In the security literature, little attention is paid to the notion of purpose. Authorization policies specify who can do what with a resource, but do not really address the use beyond the system concept of the action involved. If a doctor is authorized to make a copy of a medical test, then it becomes outside of the domain of the security system to follow how the information is used. Having policies that can travel with information and can be interpreted by other systems is an approach to such problems, but it is a complex unresolved issue (e.g., who would pay to re-architect existing systems to accommodate such a solution?).

Another issue for security policies is that there are many different policy types. Although standards exist that cover privacy policy formats, they describe a single policy type (e.g., a XML format for a general access control policy). For security policies, there are many different types—access control, firewall, network, and so on. If people are going to manage sets of different policy types, it might be useful to design similar methods for the basic functions of authoring, analyzing, and visualizing such policies. Recent policy research has focused on providing usable tools for policy template authors (Johnson et al. 2010a,b). A policy template is a policy format that enables users to create policies that covers their activities using a form provided by the organization. The template makes policy creation efficient by creating a form that will handle the range of acceptable policies and that reduces the chances of creating policies that are considered out of bounds. The policy template may be prepopulated with the policy attributes (elements of the policy) that are considered within the organization's mission.

29.4.3.7 Usability Challenges Specific to Security: Attack Defense

Much of what we present above should seem familiar to the usability practitioner in the sense that security considerations should be considered as part of the overall system use context. The choice of authentication technique in general should consider the various trade-offs in the techniques available. Biometric techniques can be considered if users are likely to accept them and if they cover the population of expected users. If passwords are used, they should be easy for users to recall and enter, but difficult to be guessed or stolen. There are some specific issues that arise because of the nature of security as a protection against attack. The fact that it is not just the user, but also that other parties with intention to do harm that must be considered, makes considering usability issues complicated in new ways.

For the majority of usability work, the goal is to make the users' primary task understandable, efficient, and effective. For security considerations, we encounter a different situation—the fact that there are people who might be trying to deceive the user and commit fraud (Conti, Ahamad, and Stasko 2005). Of considerable recent interest in the security arena is the battle against "phishing"—the misrepresentation by someone of an identity or website intended to draw the user into interactions that can be harmful to the user. A phishing attack succeeds when a user is tricked into forming an inaccurate model of the interaction and when the user takes actions contrary to their intentions. Attacks often begin with e-mails sent to potential victims, purporting to be from an individual or organization that the user has some legitimate reason for interacting with.

The issue of phishing has been viewed as a model problem for illustrating usability concerns in security (Dhamija and Tygar 2005; Miller and Wu 2005). There is an Anti-Phishing Working Group, which has been collecting and describing phishing attacks since 2005 (Anti-Phishing Working Group 2005). Analysis of the reasons why phishing attacks are so often successful points to a number of interesting usability aspects. For example, operating systems and windowing platforms that permit general purpose graphics, also allow attackers to mimic the appearance of legitimate sites. Users tend to habituate on commonly occurring warnings about submitting data over unencrypted connections, and can easily fail to notice when they are actually entering information to an insecure site. Because organizations can invest heavily in having their names and logos associated with trust, attackers can take advantage of this association by simply convincing the user that they represent a trusted organization. Because security is generally a secondary goal for users, there is only so much attention end users are willing to expend to ward off attempts at fraud.

End users are not alone in trying to prevent attacks on systems. For people working in organizations in which there are system administrators, the usability problem is partly moved from the end user to the administrator. Here, the focus shifts to making tools for users whose primary activities include monitoring the health of the overall system. Security is one of their main tasks. Intrusion detection (ID)—the problem of detecting computer attacks in a timely manner is one of both great difficulty and utmost importance. Finding specific evidence of attack activity in the enormous number of potentially relevant alerts, packets, operating system events, and user actions presents an almost overwhelming task for an ID analyst. Visualization tools and techniques can be used to increase the effectiveness of ID analysts by more fully exploiting their visual cognition abilities. Recent work has identified promising avenues for research in visualization for ID (e.g., Goodall et al. 2005). Goodall et al. present a user-centered visualization based on understanding of the work of ID and the needs of analysts derived from the first significant user study of ID. Consistent with good information visualization practice, their tool presents analysts with both "at a glance" understanding of network activity, and low-level network link details. Results from preliminary usability testing show that users performed better and found easier those tasks dealing with network state in comparison with network link tasks.

29.4.4 Summary of Human–Computer Interaction Security Research Results and Recommendations

We would all like our information to be secure; that is, we would like our systems to be resistant to access or modification from individuals whom we do not authorize to have access to them. This requires that users be aware of security mechanisms even though they would generally prefer not to. The design challenge is to make security appropriately usable—that is, to make it as easy to use as possible for the

TABLE 29.2
Design Factors Impacting Usable Security

Automatic Security versus User-Controlled Security

- Security is not generally the main user task focus for personal information management (PIM).
- "Risk Management" is not a topic that end users explicitly understand.

Authentication Mechanisms

- Access to resources is controlled by knowledge of who the user is.
- Identification relies on authentication through one (or more) mechanisms.
- Mechanisms have usability/security trade-offs.
- Can be "something I know" or "something I have."

Passwords

- Usually textual, but can be other such as graphical.
- A form of "something I know" authentication and users will forget passwords.
- Users will select passwords that are not optimally secure.

Biometrics

- A form of "something I have" authentication.
- Not all users might have a characteristic (e.g., fingerprint).
- Recognition technology is involved, so errors and attacks are possible.

Attack Considerations

- Complete design requires consideration of possible attacks.
- Anti-phishing approaches.
- Visualization for system administrator tools.

Policy Management

- Complete design requires consideration of how policies are authored, evaluated, maintained, and executed.
- Policies have much in common with rule-based system issues—they can be hard to maintain, people can have difficulty knowing what they do, and so on.

intended users, whereas making it as difficult as possible to circumvent for the potential attacker. This means that security administration and user involves trade-offs that need to be evaluated in the design of a system. Absolute security is a myth—we need to understand that it is appropriate levels of security we are looking for.

Some of the aspects of security that researchers have investigated are mentioned in Section 29.4.3 and summarized in Table 29.2. We think that serious consideration of the trade-offs between usability and security has just begun, and that much work remains. Although initial approaches have involved hiding security controls from users, we believe that this needs to be done in balance with giving the user the ability to control the level of security required.

29.5 VIEWS ON TRUST

As with privacy and security, trust is a term that can be used in many ways, and this contributes to confusion and difficulty in making progress in advancing IT systems. Trust is related to privacy and security, but differs from these concepts in that it is not an objective measure. Trust is based on the perception that the person, organization, or system one is dealing with is reliable and will act in a predictable manner. In our scenario, Mary has to make many decisions in which trust is a factor. In this discussion, we will explore factors that have been found to influence trust of online commercial entities, government organizations, and social networking sites. In particular, we will discuss the relationship between trust and risk, the factors influencing the development of trust, the relationship between trust and social networks, trust and personalization, and trust and intercultural collaborations. For the purposes of this discussion, we will assume that the privacy and security protections discussed earlier in this chapter are in place.

The relationship between trust and IT systems is complex and multifaceted. For example, users can have different perspectives on how much they trust their systems, trust the organizations they are interacting with through their systems, and trust in the provided security mechanisms that are intended to protect their privacy. There are many definitions of trust and none of them satisfies every use. These definitions vary across academic domains and with context. When dealing with e-commerce and other computer applications, the user is deciding not only to trust the individual or organization, but also the technology implementation provided by the individual or organization. Kuhlen (1998, p. 3) defined trust as "allowing us to act as if we have perfect knowledge about the reliability of that entity even if we do not." Marsh and Dibben (2003, p. 466) defined trust as concerning "a positive expectation regarding the behavior of someone or something in a situation that entails risk to the trusting party." Jøsang and Presti (2004, p. 1) defined trust as "the extent to which one party is willing to depend on somebody or something, in a given situation with a feeling of relative security, even though negative consequences are possible."

Given the many attempts at privacy definitions and the recognition of the importance of trust to the success of all types of online enterprises (Hoffman, Novak, and Peralta 1999), researchers have also attempted to model trust. Schultz (2006) proposes a model of trust and trustworthiness. This model explores how an individual's trust in another entity's trustworthiness evolves over time depending on the outcome of different and repeated situational contexts. Grabner-Krauter, Kaluscha, and Fladnitzer (2006) built a model based on characteristics of the context, the truster, and the trusted party or object. They point out that trust only is required when there is risk, and that an individual's willingness to trust and other personal characteristic affect whether or not they will develop trust. Additionally, they point out that several characteristics of the trusted party, such as dependability, predictability, benevolence, and integrity, also affect the development of the trust relationship.

There is one area of online activity in which users trust decisions increasingly can have immediate consequences if the wrong decision in made—determining if an e-mail is from a legitimate source or is a phishing attack. Kumaraguru, Acquisti, and Cranor (2006) proposed a trust model that is tailored to capture differences in expert and nonexpert recognition of these potential phishing attacks. Their model includes representations of the context or state of the world, signals that the user can detect from the target e-mail, actions that the user may take, and the user's well-being. They found that security experts were much better at detecting a range of signals from the e-mails that indicate potential phishing attacks and using this information to avoid actions that would be counter to their well-being.

All of the definitions and models share the concept that trust is about allowing us to feel comfortable allowing another entity, such as an organization or an organization's computer system, to take one or more actions for us even though there is risk that we could be harmed in some way. In other words, in the online world, the trusting party believes that the trusted organization has implemented privacy and security so that it will protect the PI of the individuals interacting with it.

29.5.1 Trust and Risk

One element that all the definitions and models have in common is that trust cannot exist without risk (Grabner-Krauter, Kaluscha, and Fladnitzer 2006; Schultz 2006). If a situation is without risk, then there is no need for trust. As Patrick, Briggs, and Marsh (2005, p. 81) pointed out, "Trust is intimately associated with risk," and people evaluate the degree of risk in many situations both online and offline. For example, Mary must evaluate risk at several points in our scenario. Mary's medical records contain sensitive PI and sharing them with a new doctor's office staff does presents risks. She must worry about both whether the staff in the medical office will treat any knowledge they learn about her appropriately and whether the records will be secure in their computer system. In our scenario, she reads and consents to the privacy policy before giving her medical records to her new doctor. Although the use of the doctor's MyMedicalFile system is very convenient for patients, it also raises privacy and security risks since it makes parts of Mary's medical records available over the Internet. If the security measures that the practice has put in place were somehow breached, unauthorized people might have access to these records. Mary must also consider the risks of using a social networking site to communicate with friends and family. Again, although this service provides convenient methods for Mary to disseminate information and receive the emotional support she needs from friends and family, it is possible that she will face negative consequences if people she did not intend see her information on this site. In all of these cases, Mary must decide whether she trusts the other entities and their IT systems to enforce her privacy choices and to provide effective security to protect her information.

In the online world, decisions like those Mary faced in our scenario require different skills to judge the trustworthiness of the entities she is dealing with than in the physical world. People can become victims of phishing attacks and identity theft if they trust too much. However, as Friedman, Kahn, and Howe (2000) point out, there are costs to trusting too little. Many opportunities can be missed if users do not take advantage of the online information and services available to them. Friedman et al. discussed several issues that make online trust and risk decisions more difficult than face-to-face transactions such as the anonymity of online transactions, the difficulty in assessing the reliability and security of technology, and misleading statements and images. Patrick, Briggs, and Marsh (2005) referenced work by Chaiken (1980) that described two strategies used by individuals depending on the perceived level of risk involved. Where people are not highly involved with the decision, they often make decisions based on appearances, and when they are more deeply involved, they use strategies that are more systematic (Chaiken 1980). This suggests that the degree to which users believe they are at risk will determine the strategies they use to assess the credibility of a website or system and the organization behind it. This model is similar to the three-stage model proposed by Sillence et al. (2004). In the first stage of their model, users make quick decisions based on appearance, which is similar to Chaiken's description of when people are not highly involved. In the second stage, Sillence et al. (2004) described the user perceptions of the credibility of the information, which is similar to the systematic strategies that Chaiken described when people are more involved.

29.5.2 Factors Influencing Trust

In our scenario, Mary decides to trust the doctor's office and its associated technology and the social networking site she used and the websites that she consulted for information. An important research question is what leads an individual user to trust or not trust a website or an online service. Brodie, Karat, and Karat (2004) showed that greater degrees of transparency and control regarding the use of PI do increase website visitor trust in the domain of IT equipment. In addition, to control one's own data and the transparency of its use, researchers have found that individuals' familiarity with a website and the perceived credibility and quality of the site also affect trust. Fogg et al. (2001) found that website visitors' perceptions of the credibility of a website were most influenced by the degree to which it was connected to a known organization in the real world and whether it seemed well designed and implemented.

Building on this research, Fogg et al. (2003) conducted a study with 2500 participants to understand factors that influence the perceived credibility of a website. They found 18 factors that influenced credibility. The biggest factor by far was the design of the appearance of the website. This was followed by the information design or structure, information focus, perceived motivation of the website owner, perceived usefulness of the information, accuracy of the information, name recognition and reputation of the website, the tone of the writing, the identity of the sponsor, functionality on the site, customer service, past experience with the site, clarity of the information, performance on a test, readability, and affiliations (Fogg et al. 2003). They attribute the importance of the design look of the site to the fact that it is very prominent when a visitor looks at a website. This supports Fogg's theory that credibility is based on noticing a feature and then interpreting its quality to create judgment (Fogg 2003). This theory is posited as a reason why there are apparent differences in findings relating to the affect of privacy guarantees (i.e., privacy statements and privacy seals) on websites (Palmer, Bailey, and Faraj 2000; Turow 2003). Palmer et al. found that privacy seals and statements did contribute to user's trust of website, whereas Turow found that website visitors do not read privacy statements or understand the meaning of seals and that there is a need to educate the public regarding privacy issues and the web.

Lumsden (2009) looked at the factors that effect initial trust in a commercial website and found that the important factors for trust were similar to those found by other researchers. These included the presence of security certifications such as Verisign, customer recommendations and testimonials, privacy and company policy statements, and contact information. When particular security features of the site were highlighted for the user (i.e., use of Secure Sockets Layer), these were also rated highly, but otherwise not. This suggests that many users do not notice them without aid. They found that although the perceived professionalism of the site design was important, it was more important when the appearance and quality of the products being sold were harder to judge online. For example, they found it was more important on clothing websites than bookseller sites.

Given that familiarity is a factor in establishing trust, Zhang and Ghorbani (2004) have identified several factors that influence familiarity, including: the individual's knowledge of similar services (prior experience), the number of times she has visited the particular website before (repeated exposure), the length of each visit to the website (level of processing), and the interval of time between visits (forgetting rate) (Zhang and Ghorbani 2004). Corritore, Kracher, and Wiedenbeck (2003) have created a model using similar factors as influencing trust in the online world. In this model, trust is determined by both external factors and the user's perception of the application or website. The external factors include the user's propensity to trust and prior experience in similar situations and their perception of the credibility of the website, how easy it is to use, and the degree of risk involved. All of these researchers' findings suggest that Mary's decision to trust the website was based on the degree to which she felt she had control of the information about her, the professionalism of the doctor's office, its website, the social networking service she used, and the informational websites, the reputation of each organization, and past experiences she has had.

Hartmann, De Angeli, and Sutcliffe (2008) studied the effect of positive and negative framing of information provided a priori to users on the usability, service quality, and look and feel of websites. They found that although all the participants were given the same information, they responded more positively if the information was put in stated in a positive way ("90% of users experience the website as easy to use") rather than a negative way ("10% of users experience difficulty using the website"). The results of this study suggest that positive recommendations Mary received regarding the doctor from friends and colleagues may have also contributed to her trust decisions.

After Mary received her diagnosis, she decided to research her condition online before her first appointment with a specialist. For each website that Mary considered, she had to decide whether the information provided was credible and trustworthy. In this case, she was not as worried about someone trying to steal her PI, but on whether or not to trust information that might affect her health decisions. Briggs, Simpson, and De Angeli (2004) looked at trust in a different type of informational website—advice websites. They have developed a model of how people determine whether or not to trust the advice they receive from websites. The authors developed a 22-point scale designed to break down trust into a set of judgments that can be measured. This list includes: whether the user perceives the information to be prepared by an expert or a knowledgeable source, whether comments from other users were available on the site, if the site was owned by a known and respected company, if they had to wait a long time on the site, if different options were suggested by the site, if the site was perceived as hard to use, whether the user felt involved in how the site constructed the advice offered, if the site was perceived to be interactive, if the advice was tailored to the user, if the reasoning was explained to the user, if there was an opportunity to contact a human, if the advice appeared to be impartial, if the advice was perceived to be good and the user trusted it, and if the site behaved in a predictable way (Briggs, Simpson, and De Angeli 2004). This study supported an earlier study that found trust was influenced by the perceived credibility of the source of the information, whether the site was personalized so that information was tailored to the individual, and whether the site is operated in a predictable way (Briggs et al. 2002). Although many have found that trust affects the use of personalization, Briggs et al. found that the presence of personalization can contribute to trust if it allows the visitor to feel that the site is tailored to their needs.

Sillence et al. (2004) had similar results when studying trust of information on health-related websites. They have proposed a three-stage model of trust in which visitors make initial, rapid assessment of a website based on the design and appearance of the site and then do a more systematic evaluation in which the credibility of the site and the degree to which it appears to be personalized to their situation becomes more important. The participants were looking for information written by or for people in situations similar to their own. The third stage of this mode addresses the maintenance of the trust relationship over time and is left as future work in this paper.

One currently popular source of information on the web is Wikipedia. This wiki provides information on a very large range of topics including illnesses and other health-related issues. Information on the site can be suspect because it is provided by members of the general public who may or may not have expertise on the subject. Suh et al. (2008) have developed an interactive visualization to show the information creation and revision history of each article to help users better determine the quality of the information. The initial results showed the users felt this tool gave them important information about the quality of information within Wikipedia. Pirolli, Wollny, and Suh (2009) followed up on the previous research by studying the effect of their visualizations on both people who tended to be more or less skeptical of Wikipedia. They found that both groups trusted the information appropriately more often when using the visualization.

Patrick, Briggs, and Marsh (2005) provide a thorough overview of many issues related to trust, including credibility of information on a website, familiarity with the website, and external factors such as prior knowledge and disposition to trust. They also describe how trust is developed slowly over time, but a bad experience or even a processing error can destroy that trust very quickly. They review a number of privacy models that have been operationalized using questionnaires including Bhattacherjee's model that describes how familiarity leads to trust and to a willingness to do business with a site (Bhattacherjee 2002) and Corritore et al.'s model that is described above (Corritore, Kracher, and Wiedenbeck 2003). Patrick, Briggs, and Marsh then conclude with design guidelines for promoting trust based on the literature that they review (Patrick, Briggs, and Marsh 2005).

Many researchers who have studied the development of trust in the online world have identified the transparency, reputation, and perceived intentions of organization and their associated websites and the ability of users to control access to their information. One active area of research that can help to address these user concerns is the use, analysis, and enforcement of organizational policies. These policies can enhance an organization's ability to provide high quality and enforceable security and privacy controls. However, if the website users do not understand or are unaware of the policies or they are not enforced by the organization, they will not help to build trust. Karat and Karat (2010a,b,c) describe research into a natural language policy authoring system to allow organizations to write policies in natural language and transform it into machine-readable formats. Karat et al. (2009) build on this work by describing research into a policy framework that has the potential to reduce these concerns by transforming policies from the human understandable natural language form to the executable policy that can be executed by an enforcement engine.

Johnson et al. (2010a,b) further builds on this research with a template-based policy authoring approach with the goal of creating enforceable and understandable policies. The templates allow both policy creators and implementers to have a better understanding of the natural language policy and to ensure that the final executable policy better adhere to the original intent of the natural language policy. From a trust perspective, this research has the potential to benefit two different groups in two different ways.

First, in this research different policy creation roles were identified ranging from policy template authors who take a broad, high-level view of the organization though to specific low-level policy authors who are responsible for policies in a small portion of the organization. Many of the policy template authors that participated in the study reported concern regarding lower level policy authors who may inadvertently create policies that do not meet the intended standards of the high-level policy management. In other words, the policy template authors do not trust the policy authors to be able to consistently create policies that meet the organization's standards. The use of templates have the potential to help address this trust issue by providing guidance and limits on the policies that individual policy authors can create.

Second, policies created with well-understood templates would allow end users to also understand the policies that are in effect and to build appropriate trust models. In our scenario, this technology would, for example, allow Mary to better understand the privacy policy that she agreed to at the doctor's office or the social networking site and feel more comfortable using the MyMedicalFile system and the social networking site.

29.5.3 Trust and Social Networking

One area of online user activity that requires trust decisions is the use of social networking sites such as Facebook, LinkedIn, and MySpace. Given these sites growing popularity, this area warrants its own analysis and discussion. Social networking sites offer users many advantages. They provide easy methods of networking with new and existing colleagues from a professional standpoint and convenient ways of finding, connecting, and communicating with old friends. In our scenario, Mary uses an online networking site to let colleagues, friends, and family who live in many different geographies know about her medical condition. This has the potential to be a powerful means of support for Mary through her illness, but it does come with associated risks. These risks range from someone gaining enough information to commit identity theft or even target a person for a crime in the physical world to people seeing PI about Mary that she did not intend for them to see and using this information against her perhaps by limiting her career advancement based on fear of her cancer returning.

There have been many accounts of negative results from sharing too much PI on social networking sites without proper thought to either privacy settings or the information posted on the sites. Acquisti and Gross (2006) reported on disciplinary actions taken against college students when pictures on Facebook showed them violating their dormitory's alcohol policy. They studied the use of Facebook by college undergraduates, graduate students, facility, and staff. At the time this paper was presented in 2006, Facebook was mainly used by college and high school students. They found that the majority of their participants reported that they were concerned about privacy and did understand how to control access to the data using the controls provided by Facebook, but their behavior differed from their stated concerns in the amount of information they disclosed. They also found that there was a sizeable minority who did not understand how to use the privacy profile functionality within Facebook to control access to their data.

Strater and Lipford (2008) also reported many potential negative consequences to revealing too much information through social networking sites, including embarrassing situations and blackmail, physical and online attacks, and the use of data by law enforcement for the investigation of crimes. They also studied students at a major university. They found that students either made everything open and planned to protect privacy using self-censorship or set all their profiles to be friends-only so that they could control the audience. New users seemed very aware that many people could be looking at their data and therefore often had an appropriate degree of trust in the system, but as time went on this changed. As users used the system to correspond only with friends through Facebook, they came to feel that the audience for their data was limited to only their friends. This has the potential to create an inappropriate degree of trust in the system. The researchers also found that many students only set their profiles once and then rarely went back to update it. They also often did not understand the implications of all the settings so they were revealing more information than they intended to.

Stutzman and Kramer-Duffield (2010) studied students at a major university in the United States who chose to use friends-only privacy settings on Facebook. First, they consider many implications of the use of the friends-only setting. Although this setting provides control of a user's audience, it does limit the power of a social networking site by greatly reducing the potential to develop new relationships with people one does not know in the physical world. Secondly, there are still ways that information about an individual may be available to a wider audience than the user realizes if others post information about them or post pictures in which they are tagged. The only method for controlling this type of disclosure is through negotiations with ones' friends. In this research, they also studied what characteristics were associated with the use of the friends-only networks. Females who had large networks and who were likely to discuss and negotiate privacy issues with friends and who were aware that casual acquaintances on campus may be looking at their information were most likely to use friends-only privacy settings. They also noted that the students were not so aware or concerned about outsiders (e.g., potential employers, law enforcement, school administration, etc.) viewing their information. The researchers highlight the need for greater education regarding privacy and the use of the privacy profiles in social networking sites such as Facebook.

Although many researchers are considering the privacy and trust implications of the use of social networking sites, most have simply proposed increased education to help users understand how to control their privacy settings and create an appropriate level of trust. Mannan and van Oorschot (2008) have proposed an approach that allows users to control access and increase their trust in a user-friendly manner. They suggest that many of the social network site users already have a "circle of trust" established with their instant messaging (IM) contact list. They suggest that this list could be utilized to automatically limit access to the same list. One major advantage of this scheme is that it would require little set-up work on the part of the social network user. It is clear that over time privacy perceptions by both individuals using social networking sites and the social networking sites themselves are changing even as usage grows. The default policy of at least one of the major social networking sites has been becoming less private over time (Opsahl 2010), although it is not clear that users are aware and sensitive to these changes. Research must continue to evolve in this growing and changing area.

29.5.4 Trust and Personalization

Another important area in dealing with online entities in which trust has important implications is personalization. In the last decade, many e-commerce companies have invested in personalizing the e-commerce websites to encourage business. However, personalization is only useful if users are willing to share data with the website, and this has not always been the case. This requires that users trust that owners of the website will not misuse their data. Hagan (2000) found that website visitors often show their distrust of Internet websites by disabling cookies on their computers and entering incorrect information into online forms. Many researchers have studied what aspects of an application or website affect user's willingness to use personalized features of websites. The Office of the Federal Privacy Commissioner of Australia (2000) conducted a large study of the factors that affect whether individuals are willing to share information with a business or other organization. They found that it is important to individuals that they understand how their data will be used and that they have control over who has access to it. Welty and Becerra-Fernandez (2001) examined how information technologies can reduce transaction costs and increase trust by increasing the symmetry of information available to both sides of the transaction in a business-to-business setting. Likewise Brodie, Karat, and Karat (2004) showed that greater degrees of transparency and control regarding the use of PI do increase willingness to share data to get access to personalized features of websites in the IT domain. Karat and Karat (2010a,b,c) further expand on this research by discussing that in addition to transparency and control of data, functionality that made the customer more effective in their own job also increased potential trust. They found that e-commerce website customers were more willing to share personal data when provided with functionality that was truly useful to them. Examples of these technologies include tracking previous customer purchases and providing compatibility information on potential new purchases, providing automatic support alerts, and tracking transactions. These findings suggest that website users' trust of an e-commerce website increases when the website allows the user to control their own PI on the site and provides value to them in their jobs or other time-critical areas of their lives.

Cranor (2003) discusses several risks that potential e-commerce customers perceive when using a personalized website. These include unsolicited marketing, potential price discrimination based on a profile, and personalization functionality, making inferences about them that are either incorrect/frustrating or correct but that the user would rather not have known by others. There are many different entities that potential website users may not want to have access to these inferences including the company itself, other users of their computer, hackers who steal information from the e-commerce company, and law enforcement. Cranor suggests several approaches to addressing these concerns including pseudonymous profiles in which users' real names are not stored, encrypted, client-side profiles so that not only is the information not stored on the enterprise's server, it is also not readable by other users of the client computer, tasked-based personalization that is deleted at the end of one session, and user control of data. Although each of these approaches has implementation costs and limitations, they provide a set of methods that could be useful depending on the specifics of the website and the situation.

Overall, trust and personalization have a complex relationship in which the presence of personalization functionality that can provide website visitors with information tailored to their situations and needs has been found to lead to a greater degree of trust on a website. At the same time, website visitors must trust the website and the organization behind it enough to share PI if the functionality is going to be effective and provide value both to the user and organization. Research has shown that website visitors are willing to share more PI when they understand how and by whom their data will be used and when they have the ability to control how it is used, which suggests that these are important factors for designers to keep in mind when designing their websites.

29.5.5 Trust and Intercultural Collaboration

Another area that deserves special discussion is the need for trust in intercultural collaborations. In the modern world, teams are often separated by large geographic distances and cross country and language borders. These distances and differences can lead to misunderstandings because team players do not always share cultural understanding to the same degree as do teams that are co-located and from the same culture. These misunderstandings can negatively impact trust that may be developing between culturally diverse partners. Many researchers have studied how to identify cultural differences that may affect productivity to improve trust issues and facilitate collaboration in many different domains. Dalberg et al. (2006) have developed a framework for identifying risks due to cultural differences on large software development projects within the European Union. The goal of their research is to understand areas that might lead to misunderstandings, inefficiencies, and loss of trust during large projects. Their framework considers the strengths and weaknesses of each team to assess risks based on contextual, cultural, and collaboration process dimensions.

Quinones et al. (2009) studied intercultural collaboration among students from different universities located in the United States, Brazil, Israel, and Turkey working on a building project to understand how their different mental models affected the team dynamics. They identified many differences in understanding that led to bad feelings and lack of trust between team members. These included differences in the perspective on the meaning of a deadline as a hard and fast time to be finished by the U.S. students versus something a bit more flexible by the Brazilian students and differences in the willingness to share some PI before starting the business of a meeting between U.S. students and Israeli students. This research went a step further than the previous project in that it concluded with recommendations for the development of collaborative tools to address these issues.

Karat et al. (2009) extend intercultural collaboration research by developing a policy framework that can be used to create tools to address the issues found. They describe research into context-sensitive policy management for collaborative mission planning. Although the current research is centered in the military domain, the teams involved have studied planning in a range of commercial domains and the proposed policy framework is generalizable to many other domains. They have combined research techniques such as Contextual Inquiry and Cultural Network Analysis to understand how policies are created and implemented from the high-level natural language level to the low-level implementation. The goal is to create a policy framework that not only performs the necessary transformations, but also detects potential conflicts between partners and even within a single organization's policies and suggests possible resolutions to those conflicts. A context-sensitive framework of this type has the potential to identify possible misunderstandings before they become serious and help to resolve them and, therefore, foster trust between the culturally diverse organizations.

29.5.6 Summary of Human–Computer Interaction Trust Research Findings and Recommendations

Trust is necessary to function in an uncertain world. The Internet continues to grow in importance in our lives. Although the traditional e-mail, IM, and informational websites continue to be important, other Internet services are growing in usage every day. These include e-commerce, social networking, and tools that allow widely distributed teams to collaborate across time, distance, and cultural differences. Trust is an important consideration in users' acceptance of all of these online services. Although this online functionality provides a great deal of value to society, it does come with serious risks and it is important that users develop an appropriate level of trust. Although there has been a great deal of research on why and when people trust in the online world, there is more to do. HCI researchers and practitioners can help by continuing to build an understanding of what leads individuals to trust a website or an application and to drive the research into new technologies that can help to allow users to develop appropriate trust models of the online websites and functionality that they use. Table 29.3 contains a summary of the factors

TABLE 29.3
Factors Influencing Online Trust in Reviewed Research

<table>
<tr><th colspan="4">Online Trust Issues Spanning All Research Areas</th></tr>
<tr><td colspan="4">• Perceived credibility of the website or application
• User familiarity with a website or application
• Predictability of the behavior of the website
• External factors (propensity of the user to trust and users’ prior experiences in similar situations)
• Ease of use
• Perceived degree of risk in dealing with the site
• Use of policy templates can address policy template creators’ concerns regarding low-level policy authors creating high-quality policies
• Policy framework and templates allow end users to better understand policies in place</td></tr>
<tr><th>Advise Websites</th><th>E-Commerce/Personalization</th><th>Social Networking</th><th>Intercultural Collaboration</th></tr>
<tr><td>• Perceived quality of information provided
• Professionalism of the site design
• Reputation of entity</td><td>• Professionalism of the site design
• Reputation of entity
• Site personalized to provide value to user
• Ability to control information shared with site
• Degree of transparency regarding use of shared information</td><td>• Ability to control information shared with site
• Degree of transparency regarding use of shared information
• User perception of information audience</td><td>• Identify and resolve misunderstandings
• Assist with cultural education</td></tr>
</table>

identified in the research cited in this chapter that have been found to influence trust in the different areas of the online world to date. HCI practitioners and website designers need to understand the factors that influence trust and use this knowledge as they design websites and applications.

29.6 CONCLUSIONS AND DIRECTIONS

The intersection of HCI, privacy, security, and trust is emerging as a critical area for research amid the backdrop of recent world events. Information is being improperly disclosed in ways that causes real harm to people through identity theft. Organizations are concerned about theft of proprietary data and documents. Citizens are becoming increasingly concerned about the use of the vast amounts of data that organizations and governments have about them, and are increasingly concerned about how that information might be used. The rapid advancement of the use of IT in industry, government, and academia makes it much easier to collect, transfer, and store sensitive information of a variety of types around the world. IT Professionals are faced with technical challenges that result from inadequate consideration of security and privacy issues in architecting current information systems, which, if not addressed, will result in a significant deterioration of the trust that people have in IT. And while there certainly is considerable research in this area, particularly on the security side, it is becoming increasingly clear that really making our systems secure and enabling appropriate attention to privacy issues will require more than just a technology focus. Although usability has been identified as a major challenge to moving the results of security and privacy research into use in real systems, it is not clear that this challenge has reached the interdisciplinary researchers needed to carry out the complex agenda of work.

We see privacy as a complex social issue concerned with individuals’ rights to know what information is collected about them and how it might be used. We do not see a solution for appropriate use of information through identification of a single policy or approach for information privacy, but argue as others have that consideration of privacy and tools to enable it need to develop hand in hand (e.g., Iachello and Abowd 2008). Technology is forcing us to rethink how we think about privacy—how much of it we want and how we can control it. This is not a simple user productivity issue, nor is it an issue we expect to see “resolved.” We fully expect that we will see rapid changes in this arena over the next decades.

We see security as an issue that will also evolve rapidly as technology develops. As we rely more and more on technology in our daily lives, we need to know that the technology functions as expected and is not subject to malicious modification in behavior. This will continue to be dealt with as both a technical issue in which guidance is to minimize the human role in the development of secure systems, and as a socio-technical issue in which the users of the technology play an important role in keeping the systems secure. The role of HCI research in security is less clearly defined than it is for privacy in that it is easier to consider our information privacy “requirements” and to imagine technology that would enable those requirements than it is to consider our information system security “requirements” and the technology that would enable those.

Trust differs from privacy and security in that it is more about individual’s perception of the risks of a situation than about measurable objectives regarding the protection of data and systems. Historically, this difference has driven trust research directions toward understanding trust issues in the online world and to the development of trust models that explore the characteristics of both users and systems that either trust or do not trust technology and the organizations providing it. In recent years, HCI researchers have begun to broaden the trust research in many areas of the online world by creating recommendations and prototypes of technologies to help address the trust issues and allow users to develop an appropriate level of trust in systems, the organizations behind the systems, and even other end users that they may interact with through the technology. In the future, more HCI research is needed to ensure that trust-enhancing technologies that are created and implemented really meet users' needs in these areas. As many researchers have pointed out, trust in online services is necessary for the success of all facets of the online world, from e-commerce to social networking to e-medicine and e-government.

Deciding how we can design usable and effective privacy and security technology capabilities for the future includes legal, cultural, business, social policy, human performance, and practical dimensions. Since at its core, the goals of HCI and user-centered design are to understand the user, the user's tasks or goals, and the social and physical context in which the user is interacting with the system, all of these dimensions can be considered as within the scope of creating usable and effective privacy and security solutions.

HCI research in privacy, security, and trust are critical areas of focus in today's world. Communication and collaboration is occurring between researchers, academics, and practitioners with expertise in one or more of the areas of privacy, security, trust, and HCI. This emerging community is beginning to work together to incorporate HCI in the design and development process for usable and effective privacy and security solutions. This chapter has reviewed and synthesized the research on HCI and privacy, security and trust, calling out many key areas in need of future research. Please join the emerging professional community focusing on the worthy effort to create usable and trusted privacy and security solutions for end users and organizations worldwide.

REFERENCES

Ackerman, M., L. Cranor, and J. Reagle. 1999. Privacy in e-commerce: Examining user scenarios and privacy preferences. In *Proceedings of the ACM Conference on Electronic Commerce*, 1–8. Denver, CO: ACM.

Ackerman, M., and S. Mainwaring. 2005. Privacy issues in human-computer interaction. In *Security and Usability: Designing Secure Systems That People Can Use*, ed. L. Cranor and S. Garfinkel, 381–400. Sebastopol, CA: O'Reilly.

Acquisti, A., and R. Gross. 2006. Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook. Springer eBook. Berlin: Springer.

Adams, A., and A. Blanford. 2005. Bridging the gap between organizational and user perspectives of security in the clinical domain. *Int J Hum Comput Stud* 63(1–2):175–202.

Adams, A., and A. Sasse. 1999. Users are not the enemy: Why users compromise security mechanisms and how to take remedial measures. *Commun ACM* 42(12):40–6.

Anderson, R. 1994. Why cryptosystems fail. *Communications of the ACM* 37(11):32–40.

Anti-Phishing Working Group. 2005. APWG Phishing Archive. http://anti-phishing.org/phishing_archive.htm (accessed July 23, 2010).

Ashbourn, J. 2000. *Biometrics: Advanced Identity Verification.* London: Springer Verlag.

Associated Press. 2003. Pentagon Spy Office to Close. *Wired News*. http://www.wired.com/news/print/0,1294,60588,00.html (accessed September 25, 2003).

Balfanz, D., G. Durfee, R. E. Grinter, D. Smetters, and P. Stewart. 2004a. Network-in-a-Box: How to set up a secure wireless network in under a minute. In *Proceedings of the 13th USENIX Security Symposium,* 207–22. Berkeley, CA: USENIX Association.

Balfanz, D., G. Durfee, and D. K. Smetters. 2005. Making the impossible easy: Usable PKI. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 319–34. Sebastopol, CA: O'Reilly.

Balfanz, D., G. Durfee, D. K. Smetters, and R. E. Grinter. 2004b. In search of usable security: Five lessons from the field. *IEEE Secur Privacy* 2(5):19–24.

Beckles, B., V. Welch, and J. Basney. 2005. Mechanisms for increasing usability of grid security. *Int J Hum Comput Stud* 63(1–2):74–101.

Begole, J., J. C. Tang, R. B. Smith, and N. Yankelovich. 2002. Work rhythms: Analyzing visualizations of awareness histories of distributed groups. In *Proceedings of CSCW 2002*, 334–43. New Orleans, LA: ACM Press.

Bernaschi, M., E. Gabrelli, and L. Mancini. 2000. Operating system enhancements to prevent the misuse of system calls. In *Proceedings of the 7th ACM conference on Computer and communications security*, 174–83. New York: ACM Press.

Bellotti, V. 1996. What you don't know can hurt you: Privacy in collaborative computing. In *Proceedings of the HCI Conference on People and Computers XI*, 241–61. London: Springer-Verlag.

Bellotti, V., and A. Sellen. 1993. Designing for Privacy in Ubiquitous Computing Environments, In *Proc European Conference on Computer-Supported Cooperative Work ECSCW'93*, 77–92. Norwell, MA: Kluwer Academic Publishers.

Bertino, E., C. Brodie, S. Calo, L. Cranor, C.-M. Karat, J. Karat et al. 2009. Analysis of privacy and security policies. *IBM J Res Dev* 53(2–3):1–18.

Bhattacherjee, A. 2002. Individual trust in online firms: Scale development and initial trust. *J Manage Inf Syst* 19(1):213–43.

Briggs, P., B. Buford, A. De Angeli, and P. Lynch. 2002. Trust in online advice. *Soc Sci Comput Rev* 20(3):321–32.

Briggs, P., B. Simpson, and A. De Angeli. 2004. Personalization and trust: A reciprocal relationship? In *Designing Personalized User Experiences in e-Commerce,* ed. C.-M. Karat, J. Blom, and J. Karat, 39–56. Dordrecht, Netherlands: Kluwer Academic Publishers.

Brodie, C., D. George, C.-M. Karat, J. Karat, J. Lobo, M. Beigi, X. Wang, S. Calo, and D. Verma. 2008. The coalition policy management portal for policy authoring, verification, and deployment. In *Proceedings of the IEEE Policy 2008 Conference*, 247–49. New York: IEEE.

Brodie, C., C.-M. Karat, and J. Karat. 2004. Personalizing interaction. In *Designing Personalized User Experiences in e-Commerce*, ed. C.-M. Karat, J. Blom, and J. Karat, 185–206. Dordrecht, Netherlands: Kluwer Academic Publishers.

Brodie, C., C. Karat, and J. Karat. 2006. An empirical study of natural language parsing accuracy of privacy policy rules using the SPARCLE Policy Workbench. In *Proceedings of the Symposium on Usable Privacy and Security.* ACM Digital Library, submitted for publication. New York: ACM.

Brostoff, S., and A. Sasse. 2000. Are passfaces more usable than passwords? (Eds.) In *Proceedings of HCI 2000 Conference on People and Computers XIV—Usability or Else!* 405–24. London: Springer.

Buffett, S., M. W. Fleming, M. M. Richter, N. Scott, and B. Spencer. 2004. Determining Internet users' values for privacy information. In *Proceedings of the Second Annual Conference on Privacy, Security, and Trust*, 79–88. Toronto, Canada: National Research Council of Canada.

Caloyannides, M. A. 2004. Speech privacy technophobes need not apply. *IEEE Secur Privacy* 2(5):86–7.

Chaiken, S. 1980. Heuristic versus systematic information processing and the use of source versus message cues in persuasion. *J Pers Soc Psychol* 39:752–66.

Conti, G., M. Ahamad, and J. Stasko. 2005. Attacking information visualization system usability: Overloading and deceiving the human. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS '05)*. New York: ACM Digital Library.

Corritore, C., B. Kracher, and S. Wiedenbeck. 2003. Online trust: Concepts, evolving themes, a model. *Int J Hum Comput Stud* 58(6):737–58.

Coventry, L. 2005. Usable biometrics. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 175–98. Sebastopol, CA: O'Reilly.

Cranor, L. 2002. *Web Privacy with P3P*. Sebastopol, CA: O'Reilly Media.

Cranor, L. F. 2003. 'I didn't buy it for myself' privacy and e-commerce personalization. In *Proceedings of the 2003 ACM Workshop On Privacy in Electronic Society*, 111–17. New York: ACM.

Cranor, L. 2004. 'I didn't buy it for myself': Privacy e-commerce personalization. In *Designing Personalized User Experiences in e-Commerce*, ed. C.-M. Karat, J. Blom, and J. Karat, 57–74. Dordrecht, Netherlands: Kluwer Academic Publishers.

Cranor, L. 2005. Privacy policies and privacy preferences. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 447–72. Sebastopol, CA: O'Reilly.

Culnan, M. 2000. Protecting privacy online: Is self-regulation working? *J Publ Policy Mark* 19(1):20–6.

Dalberg, V., E. Angelvik, D. R. Elvekrok, and A. K. Fossberg. 2006. Cross-cultural collaboration in ICT procurement. In *The Proceedings of the 2006 International Workshop on Global Software Development for the Practitioner*, 51–9. ACM Press.

Davis, D., F. Monrose, and M. Reiter. 2004. On user choice in graphical password schemes. In *Proceedings of the 13th USENIX Security Symposium*, 151–64. Berkeley, CA: USENIX Association.

De Angeli, A., L. Coventry, G. Johnson, and K. Renaud. 2005. Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *Int J Hum Comput Stud* 63(1–2):128–52.

de Paula, R., X. Ding, P. Dourish, K. Nies, B. Pillet, D. Redmiles, J. Ren, J. Rode, and R. Silva Filho. 2005. Symposium On Usable Privacy and Security (SOUPS), Pittsburgh, PA, USA. New York, NY: ACM Digital Library.

Dewan, P., and H. Shen. 1998. Controlling access in multiuser interfaces. ACM Transactions on Computer-Human Interaction, 5(1):34–62. New York, NY: ACM Press.

Dey, A. K., and G. D. Abowd. 2000. CybreMinder: A context-aware system for supporting reminders. In *Proceedings of the International Symposium on Handheld and Ubiquitous Computing*, 172–86. Berlin: Springer.

Dhamija, R., and A. Perrig. 2000. Déjà Vu: A user study using images for authentication. In *Proceedings of the 9th conference on USENIX Security Symposium*, 4–13. Berkeley, CA: USENIX Association.

Dhamija, R., and J. D. Tygar. 2005. The battle against phishing: Dynamic security skins. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS).* New York: ACM Digital Library.

Dingledine, R., and N. Matherwson. 2005. Anonymity loves company: Usability and the network effect. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 547–60. Sebastopol, CA: O'Reilly.

Dourish, P. 1993. Culture and control in a media space. In *ECSCW'93 Proceedings of the third conference on European Conference on Computer-Supported Cooperative Work*, 125–37. Norwell, MA: Kluwer Academic Publishers.

Egelman, S., J. Tsai, L. Cranor, and A. Acquisti. 2009. Timing is everything? The effects of timing and placement of online privacy indicators. In *Proceedings of the Conference on Human Factors in Computing Systems—CHI 2009.* New York: ACM Press.

Ellison, C., and B. Schneier. 2000. Ten risks of PKI: What you're not being told about public key infrastructure. *Computer Security Journal* XVI(1):1–7.

Everitt, K., T. Bragin, J. Fogarty, and T. Kohno. 2009. Comprehensive study of frequency, interference, and training of multiple graphical passwords. In *Proceedings of the Conference on Human Factors in Computing Systems—CHI 2009*, 889–98. New York: ACM Press.

Fogg, B. J. 2003. Prominence-interpretation theory: Explaining how people assess credibility online. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2003, Extended Abstracts on Human Factors in Computing Systems*, 722–3. New York: ACM Press.

Fogg, B. J., J. Marshall, O. Laraki, A. Osipovich, C. Varma, N. Fang, J. Paul et al. 2001. What makes websites credible? A report on a large quantitative study. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2001*, 61–8. New York: ACM Press.

Fogg, B. J., C. Soohoo, D. R. Danielson, L. Marable, J. Stanford, and E. R. Tauber. 2003. How do users evaluate the credibility of websites? A study with over 2,500 participants. In *Proceedings of the 2003 Conference on Designing for User Experiences*, 1–15. New York: ACM Press.

Friedman, B., P. H. Kahn, and D. C. Howe. 2000. Trust online. *Commun ACM* 43(12):34–40.

Good, N., and A. Krekelberg. 2003. A study of Kazaa P2P file-sharing. In *Proceedings of the Conference on Human Factors in Computing – Systems CHI 2003*, 137–44. New York: ACM Press.

Good, N., J. B. Schafer, J. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl. 1999. Combining collaborative filtering with personal agents for better recommendations. In *Proceedings of the AAAI*, 439–46. Palo Alto: American Association for Artificial Intelligence.

Goodall, J., A. Ozok, W. Lutters, P. Rheingens, and A. Komlodi. 2005. A user-centered approach to visualizing network traffic for intrusion detection. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, 1403–6. New York: ACM Press.

Grabner-Krauter, S., E. A. Kaluscha, and M. Fladnitzer. 2006. Perspectives of online trust and similar constructs: A conceptual clarification. In *Proceedings of the 8th International Conference on Electronic Commerce: The new e-commerce: Innovations for Conquering Current Barriers, Obstacles, and Limitations to Conducting Successful Business on the Internet*, 235–43. New York: ACM.

Greenberg, S., and D. Marwood. 1994. Real time groupware as a distributed system: Concurrency control and its effect on the interface. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 207–17. New York: ACM.

Gutmann, P. Plug-and-Play PKI: A PKI your mother can use. In *Proceedings of the 12th USENIX Security Symposium*, 45–58. Berkeley, CA: USENIX Association.

Guttmann, P. 2003. Plug-and-play PKI: A PKI your mother can use. In *Proceedings of the 12th conference on USENIX Security Symposium*, 4–13. CA: USENIX Association Berkeley.

Hagan, P. R. 2000. *Personalization Versus Privacy*. Cambridge, MA: Forrester.

Harper, J., and S. Singleton. 2001. *With a Grain of Salt: What Consumer Privacy Surveys Don't Tell Us.* http://ssrn.com/abstract=299930 (accessed April 2, 2010).

Hartmann, J., A. De Angeli, and A. Sutcliffe. 2008. Framing the user experience: Information biases on website quality judgement. In *The Proceeding of the 26th Annual SIGCHI Conference on Human Factors in Computer Systems*, 855–64. New York: ACM.

Henning, R. 1999. Security Service Level Agreements: Quantifiable Security for the Enterprise? New Security Paradigm Workshop, 54–60. Ontario, Canada: IEEE.

Helft, M., and J. Wortham. 2010. Facebook bows to pressure over privacy. *The New York Times*, B1–6.

Herbsleb, J. D., D. L. Atkins, D. G. Boyer, M. Handel, and T. A. Finholt. 2002. Introducing instant messaging and chat in the workplace. In *Proceeding of the Conference on Human Factors in Computing Systems – CHI 2002*, 171–8. New York: ACM Press.

Hoffman, D. L., T. P. Novak, and M. Peralta. 1999. Building consumer trust online. *Commun ACM* 42(4):80–5.

Hong, J. I., and J. Landay. 2004. An architecture for privacy-sensitive ubiquitous computing. In *Proceedings of MobiSys'04*, 177–89. New York: ACM Press.

Hong, J. I., J. D. Ng, S. Lederer, and J. Landay. 2004. Privacy risk models for designing privacy-sensitive ubiquitous computing systems. In *Proceedings of the Conference on Designing Interactive Systems – DIS'04*. New York: ACM Press.

Horstmann, T., and R. Bentley. 1997. Distributed authoring on the web with the BSCW shared workspace system. *ACM Stand View*, ACM Press 5(1):9–16.

Iachello, G., and G. D. Abowd. 2008. From privacy methods to a privacy toolbox: Evaluation shows that heuristics are complementary. *ACM Trans Comput Hum Interact* 15(2):1–30.

IBM Secure Perspective. 2007. http://www-03.ibm.com/systems/i/advantages/security/rethink_security_policy.html (accessed December 3, 2010).

Inglesant, P., and A. Sasse. 2010. The true cost of unusable password policies: Password use in the wild. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2010*. New York: ACM Press.

Irvine, C., and T. Levin. 1999. Toward a Taxonomy and Costing Method for Security Services. In *Proceedings of the 15th Annual Computer Security Applications Conference (ACSAC '99)*, 183–5. IEEE.

Irvine, C., and T. Levin. 2001. Quality of security service. Proceedings of the 2000 workshop on New security paradigms, ACM Press, 91–9.

Jensen, C., and C. Potts. 2004. Privacy policies as decision-making tools: A usability evaluation of online privacy notices. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2004*, 471–8. New York: ACM Press.

Jensen, C., C. Potts, and C. Jensen. 2005. Privacy practices of Internet users: Self-reports versus observed behavior. *Int J Hum Comput Stud* 63:203–27.

Jøsang, A., and S. L. Presti. 2004. Analysing the relationship between risk and trust. In *The Proceedings of the Second International Conference on Trust Management*, ed. T. Dimitrakos, 135–145. UK: Oxford.

Johnson, M., J. Karat, C.-M. Karat, and K. Grueneberg. 2010a. Optimizing a policy authoring framework for security and privacy policies. In *Symposium on Usable Privacy and Security (SOUPS) 2010*. Redmond, WA: ACM Digital Library.

Johnson, M., J. Karat, C.-M. Karat, and K. Grueneberg. 2010b. Usable policy template authoring for iterative policy refinement. To be presented at *IEEE Policy* 2010. Fairfax, VA: IEEE.

Just, M. 2004. Designing and evaluating challenge-question systems. *IEEE Secur Privacy* 2(5):32–9.

Kahn, D. 1967. The Codebreakers. New York: Macmillan.

Kandogan, E., and E. Haber. 2005. Security administration tools and practices. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 357–78. Sebastopol, CA: O'Reilly.

Karat, C.-M. 1989. Iterative usability testing of a security application. In *Proceedings of the Human Factors Society*, 272–7. Santa Monica, CA: HFES.

Karat, C.-M. 1994. A business case approach to usability cost justification. In *Cost-Justifying Usability*, ed. R. Bias and D. Mayhew, 45–70. New York: Academic Press.

Karat, C.-M. 2005. A business case approach to usability cost justification for the web. In *Cost-Justifying Usability: An Update for the Internet Age*, ed. R. Bias and D. Mayhew, 103–42. San Francisco, CA: Morgan Kaufman.

Karat, C.-M., Brodie, and J. Karat. 2005. Usability design and evaluation for privacy and security solutions. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 47–74. Sebastopol, CA: O'Reilly.

Karat, C.-M., and J. Karat. 2010a. Case study 2: Personalization in e-commerce. In *Designing and Evaluating Usable Technology in Industrial Research – Three Case Studies,* 31–54. New York: Morgan and Claypool Publishers.

Karat, C.-M., and J. Karat. 2010b. Case study 3: Security and privacy policy management technologies. In *Designing and Evaluating Usable Technology in Industrial Research – Three Case Studies*, 57–87. New York: Morgan and Claypool Publishers.

Karat, C.-M., and J. Karat. 2010c. *Designing and Evaluating Usable Technology in Industrial Research: Case Studies in Speech Recognition, Personalization in E-commerce, and Security and Privacy.* Lectures in Human-Centered Informatics (Jack Carroll, Series Editor). NY: Morgan-Claypool.

Karat, J., C.-M. Karat, E. Bertino, N. Li, Q. Ni, C. Brodie, J. Lobo et al. 2009. Policy framework for security and privacy management. *IBM J Res Dev* 53(2):1–14.

Karat, J., C.-M. Karat, and C. Brodie. 2004. Personalizing interaction. In *Designing Personalized User Experiences in e-Commerce*, ed. C.-M. Karat, J. Blom, and J. Karat, 7–18. Dordrecht, Netherlands: Kluwer Academic Publishers.

Karat, J., C.-M. Karat, C. Brodie, and J. Feng. 2005. Privacy in information technology: Designing to enable privacy policy management in organizations. *Int J Hum Comput Stud* 63(1–2):153–74.

Karat, C.-M., J. Karat, C. Brodie, and J. Feng. 2006. Evaluating interfaces for privacy policy rule authoring. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2006*, 83–92. New York: ACM Press.

Karat, J., W. Sieck, T. Norman, C.-M. Karat, C. Brodie, L. Rasmussen, and K. Sycara. 2009. A framework for culturally adaptive policy management in ad hoc collaborative contexts. In *Proceedings of the ACM International Workshop on Intercultural Collaboration* 2009, 1–4. Palo Alto, CA: ACM.

Kobsa, A. 2002. Personalized hypermedia and international privacy. *Communications of the ACM* 45(5):64–7.

Kuhlen, R. 1998. Trust: A principle for ethics and economics in the global information society. Presented at *The Second UNESCO Congress for Informational Ethics*, Monte Carlo, Monaco.

Kumaraguru, P., A. Acquisti, and L. F. Cranor. 2006. Trust modeling for online transactions: A phishing scenario. In *The Proceedings of the 2006 International Conference on Privacy, Security, and Trust: Bridge the Gap Between PST Technologies and Business Services*. Markham, ON. NewYork: ACM Digital Library.

Kumaraguru, P., and L. Cranor. 2005. Privacy in India: Attitudes and awareness. In *Proceedings of the 2005 Workshop on Privacy Enhancing Technologies (PET2005)*, 243–58. Dubrovnik, Croatia. Berlin: Springer.

Lederer, S., J. Mankoff, and A. K. Dey. 2003. Who wants to know what when? Privacy preference determinants in ubiquitous computing. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2003*, 724–5. New York: ACM Press.

Little, L., E. Sillence, and P. Briggs. 2009. Ubiquitous systems and the family: Thoughts about the networked home. In *Symposium on Usable Privacy and Security (SOUPS) 2009*. Mountain View, CA: ACM Digital Library.

Lumsden, J. 2009. Triggering trust: To what extent does the question influence the answer when evaluating the perceived importance of trust triggers. In *The Proceedings of the 2009 British Society Conference on Human-Computer Interaction*, 214–23. Swinton, UK: British Computer Society.

Malin, B., and L. Sweeney. 2004. How (not) to protect genomic data privacy in a distributed network: Using trail re-identification to evaluation and design anonymity protection systems. *J Biomed Inf* 37(3):179–92.

Mandler, G. 1991. Your face is familiar but I can't remember your name: A review of Dual Process Theory, Relating Theory and Data. *J Exp Psychol Learn Memory Cognit* 11:207–25.

Mannan, M., and P. C. van Oorschot. 2008. Privacy-enhanced sharing of personal content on the web. In *The Proceedings of the 2008 Conference on the World Wide Web*, 487–96. New York: ACM.

Mantei, M. M., R. M. Baecker, A. J. Sellen, W. A. S. Buxton, T. Milligan, and B. Wellman. 1991. Experiences in the use of a media space. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI'91*, 203–8. New York: ACM Press.

Marsh, S., and M. Dibben. 2003. The role of trust in information science and technology. In *Annual Review of Information Science and Technology,* ed. B. Cronin, Vol. 37, 465–98.

Maxion, R., and R. Reeder. Improving user-interface dependability through mitigation of human error. *Int J Hum Comput Stud*, Volume 63, Issues 1-2, July 2005, 25–50.

Meyer, S., and A. Rakotonirainy. 2003. A survey of research on context-aware homes. In *Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*, Vol. 21, 159–68. Darlinghurst, Australia: Australian Computer Society, Inc.

Milberg, S. J., S. J. Burke, H. J. Smith, and E. A. Kallman. 1995. Values, personal information privacy, and regulatory approaches. *Commun ACM* 38(12):65–74.

Miller, R., and M. Wu. 2005. Fighting phishing at the user interface. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 275–92. Sebastopol, CA: O'Reilly.

Monrose, F., and M. Reiter. 2005. Graphical password systems. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 157–74. Sebastopol, CA: O'Reilly.

Ni, Q., E. Bertino, C. Brodie, C.-M. Karat, J. Karat, J. Lobo, and A. Trombetta. 2008. Privacy-aware role based access control. *ACM Trans Comput Logic* 16(8):1–35.

Office of the Federal Privacy Commissioner of Australia. 2000. Privacy and Business. Retrieved 2008, from http://www.privacy.gov.au.

Opsahl, K. *Facebook's Eroding Privacy Policy: A Timeline*. http://www.eff.org/deeplinks/2010/04/facebook-timeline (accessed April 28, 2010).

Palen, L., and P. Dourish. 2003. In *Proceedings of the SIGCHI conference on Human factors in computing systems—CHI 2003*, 129–36. New York: ACM Press.

Palmer, J. W., J. P. Bailey, and S. Faraj. 2000. The role of intermediaries in the development of trust on the WWW: The use and prominence of trusted third parties and privacy statements. *J Comput Mediat Commun* 5(3). http://jcmc.indiana.edu/vol5/issue3/palmer.html (accessed September 12, 2010).

Patrick, A., P. Briggs, and S. Marsh. 2005. Designing systems that people will trust. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 75–100. Sebastopol, CA: O'Reilly.

Peacock, A., X. Ke, and M. Wilkerson. 2005. Identifying users from their typing patterns. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 199–220. Sebastopol, CA: O'Reilly.

Pettersson, J. H., S. Fischer-Huebner, N. Danielsson, J. Nilsson, M. Bergmann, S. Clauss, T. Kriegelstein, and H. Krasemann. 2005. Making PRIME usable. In *Proceedings of the Symposium on Usable Privacy and Security.* New York: ACM Digital Library.

Piazzalunga, U., P. Salvaneschi, and P. Coffetti. 2005. The usability of security devices. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 221–44. Sebastopol, CA: O'Reilly.

Pirolli, P., E. Wollny, and B. Suh. 2009. So you know you're getting the best possible information: A tool that increases wikipedia credibility. In *The Proceedings of the Twenty-Seventh Annual SIGCHI Conference on Human Factors in Computing Systems*, 1505–8. New York: ACM.

Price, B. A., K. Adam, and B. Nuseibeh. 2005. Keeping ubiquitous computing to yourself: A practical model for user control of privacy. *Int J Hum Comput Stud* 63(1–2):228–53.

PRIME. 2008. https://www.prime-project.eu/ (accessed September 12, 2010).

Quinones, P.-A., S. R. Fussell, L. Soibelman, and B. Akinci. 2009. Bridging the gap: Discovering mental models in globally collaborative contexts. In *The Proceedings of the 2009 International Workshop on Intercultural Collaboration*, 101–10. New York: ACM.

Real User Corporation. 2002. *The Science behind Passfaces*. http://www.realuser.com/published/ScienceBehindPassfaces.pdf (accessed January 23, 2010).

Reiter, M., and A. Rubin. 1998. Crowds: Anonymity for web transactions. *ACM Trans Inf Syst Secur* 1(1):66–92.

Renaud, K. 2005. Evaluating authentication mechanisms. In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 103–28. Sebastopol, CA: O'Reilly.

Saltzer, J., and M. Schroeder. 1975. The protection of information in computer systems. *Proc IEEE* 63(9):1278–308.

Sasse, A., and I. Flechais. 2005. Usable security: Why do we need it? How do we get it? In *Security and Usability: Designing Secure Systems That People Can Use,* ed. L. Cranor and S. Garfinkel, 13–30. Sebastopol, CA: O'Reilly.

Schultz, C. D. 2006. A trust framework model for situational contexts. In *The Proceedings of the 2006 International Conference on Privacy Security, and Trust: Bridge the Gap Between PST Technologies and Business Services*, 50. New York: ACM.

Schneirer, B., and Mudge. 1998. Cryptanalysis of Microsoft's point-to-point tunneling protocol (PPTP). In *Proceedings of the 5th ACM conference on Computer and communications security*, 132–41. New York: ACM Press.

Senior, A., S. Pankanti, A. Hampapur, L. Brown, Y. Tian, and A. Ekin. 2003. Blinkering surveillance: Enabling video privacy through computer vision. *IBM Res Rep* RC22886 (W0308-109), 1–9.

Shankar, U., K. Talwar, J. Foster, and D. Wagner. 2001. Detecting format string vulnerabilities with type qualifiers. In *Proceedings of the 10th conference on USENIX Security Symposium*, 16–23. Berkeley, CA: USENIX Association Berkeley.

Sillence, E., P. Briggs, L. Fishwick, and P. Harris. 2004. Trust and mistrust of online health sites. In *Proceedings of the Conference on Human Factors in Computing Systems – CHI 2004*, 663–70. New York: ACM Press.

Smith, H. 1993. Privacy policies and practices: inside the organizational maze. In *Communications of the ACM*, 36(12):104–22. New York: ACM.

Spiekermann, S., J. Grossklags, and B. Berendt. 2001. E-privacy in 2nd generation e-commerce: Privacy preferences versus actual behavior. In *Proceedings of the ACM conference on Electronic Commerce*, 38–47. New York: ACM.

Strater, K., and H. R. Lipford. 2008. Strategies and struggles with privacy in an online social networking community. In *The Proceedings of the 22nd British HCI Group Conference on HCI 2008: People and Computers XXII: Culture, Creativity, Interaction – Volume 1*, 111–9. Swinton, UK: British Computer Society, Inc.

Stutzman, F., and J. F. Karamer-Duffield. 2010. Only: Examining a privacy-enhancing behavior in facebook. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems*, 1553–62. New York: ACM.

Suh, B., E. H. Chi, A. Kittur, and B. Pendleton. 2008. Lifting the veil: Improving accountability and social transparency in wikipedia with wikiDashboard. In *The Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 1037–40. New York: ACM.

Sweeney, L. 2002. k-Anonymity: A model for protecting privacy. *Int J Uncertainty Fuzziness Knowl Based Syst* 10(5):557–70.

Teltzrow, M., and A. Kobsa. 2004. Impacts of user privacy preferences on personalized systems. In *Designing Personalized User Experiences in e-Commerce,* ed. C.-M. Karat, J. Blom, and J. Karat. Dordrecht, Netherlands: Kluwer Academic Publishers.

Thomsen, D., and M. Denz. 1997. Incremental Assurance for Multilevel Applications. In *Proc 13th Annual Computer Security Applications Conference*. IEEE.

Thuraisingham, B. 2002. Data mining, national security, privacy and civil liberties. *ACM SIGKDD Explor Newslett* 4(2):1–5.

Turow, J. 2003. *Americans and Online Privacy: The System is broken*. Philadelphia: Annenberg Public Policy Center.

Wagner, D., J. Foster, E. Brewer, and A. Aiken. 2000. A First Step Toward Automated Detection of Buffer Overrun Vulnerabilities. In *Proceedings of the Network and Distributed System Security Symposium*. New York: ACM Digital Library.

Want, R., A. Hopper, V. Falcão, and J. Gibbons. 1992. The active badge location system. *ACM Trans Inf Syst* 10(1):91–102.

Weiser, M. 1991. The computer for the twenty-first century. *Sci Am* 94–104.

Welty B., and I. Becerra-Fernandez, 2001. Managing trust and commitment incollaborative supply chain relationships. In *Communications of the ACM*, 44(6):67–73. New York: ACM.

Wiedenbeck, S., J. Waters, J. Birget, A. Brodskiy, and N. Memon. 2005. PassPoints: Design and longitudinal evaluation of a graphical password system. *Int J Hum Comput Stud* 63(1-2):102–27.

Whitten, A., and D. Tygar. 1999. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, 169–84. Berkeley, CA: USENIX.

Yan, J., A. Blackwell, R. Anderson, and A. Grant, 2005. The memorability and security of passwords. In *Designing Secure Systems that People Can Use*, ed. L. Cranor and S. Garfinkel, 129–42. O'Reilly.

Yee, K. P. 2002. User interaction design for the design of secure systems. Springer Lecture Notes in Computer Science, 2002, Volume 2513/2002, 278–90. Berlin: Springer.

Yenisey, M., A. Ozok, and G. Salvendy. 2005. Perceived security determinants in e-commerce among Turkish university students. *Behav Inf Technol* 24(4):259–74.

Zhang, J., and A. A. Ghorbani. 2004. Familiarity and trust: Measuring familiarity with a website. In *Proceedings of the Second Annual Conference on Privacy Security, and Trust*, 23–8. Toronto, Canada: National Research Council of Canada.

Zurko, M., and R. Simon. 1996. User-centered security. In *NSPW96 New Security Paradigms Workshop V*. September 17–20, 27–33. Lake Arrowhead, CA, USA: ACM Press.

Part IV

Application-/Domain-Specific Design

30 Human–Computer Interaction in Health Care

François Sainfort, Julie A. Jacko, Molly A. McClellan, and Paula J. Edwards

CONTENTS

30.1 INTRODUCTION

U.S. health care expenditures reached $2.3 trillion in 2008 and is 16% of the U.S. Gross Domestic Product (BlueCross/BlueShield Association 2008). Despite such large spending, many Americans remain uninsured and do not have access to health care services. According to the Congressional Budget Office, about 43 million people, or 17%, of the nonelderly population are uninsured (Congressional Budget Office, 2007). Furthermore, while our country has the most formidable medical workforce in the world and develops and uses the most modern medical technologies, the World Health Organization (2000) rated the quality and performance of the U.S. healthcare systems as being worse than most of its counterparts in the Western world. In 2006, the United States was the biggest spender in terms of health care per capita but ranked 39th for infant mortality, 43rd for adult female mortality, 42nd for adult male mortality, and 36th for life expectancy (World Health Organization).

Chassin, Galvin, and the National Roundtable on Health Care Quality (1998) documented three types of quality problems: (1) overuse, (2) underuse, and (3) misuse. The results of an extensive review of over 70 publications covering years 1993 through 2000 provide "abundant evidence that serious and extensive quality problems exist throughout American medicine resulting in harm to many Americans" (Institute of Medicine 2001, p. 24). In its first report, To Err is Human, the Institute of Medicine (2000) reported serious and widespread errors in health care delivery that resulted in frequent avoidable injuries to patients. The Institute of Medicine (IOM) (2001) suggested four key underlying reasons for inadequate quality of care in the U.S. health care system: (1) the growing complexity of science and technology, (2) the increase in chronic conditions, (3) a poorly organized delivery system, and (4) constraints on exploiting the revolution in information technology. Since the IOM report was issued, health care reform has been highly debated. Current U.S. health care reform proposals focus on extending insurance coverage, decreasing the growth of costs through improved efficiency, and expanding prevention and wellness programs (Murray, Phil, and Frenk 2010). To improve quality of care, the American Recovery and Reinvestment Act (ARRA) of 2009 allotted $19.2 billion for Health Information Technology (HIT) including financial incentives to use electronic health records (EHRs) and grants and loans for HIT solutions (American Recovery and Reinvestment Act of 2009).

Reengineering the delivery of health care services through innovative development, application, and use of information and medical technologies can result in tremendous cost savings, improved access to health care services, and improved quality of life for all citizens. Health care professionals in the United States have recognized that both the information revolution and the biological revolution will offer tremendous opportunities—and challenges—for (re)designing the health care system of the future. They are aware of the need to better utilize new information and communication technologies and incorporate computing power into care delivery and clinical practice. They are also aware that the widely publicized biological revolution (which includes advances both in genetics and biomedical engineering) will soon bring a large number of screening and diagnostic tests as well as new treatment strategies and disease management tools. It is clear that the combination of biotechnology, computing power, information and communication technologies, distance technology, and sensor technology will make future delivery of health care in the United States unrecognizable from the care we deliver today.

Despite these acknowledgments regarding the need for HIT, implementations in the last decade have proceeded slowly. It is estimated that only 17% of U.S. physicians and 8–10% of U.S. hospitals have a basic EHR system, and even less have or use comprehensive HIT systems that reach their full potential (Blumenthal 2009). Reasons contributing to this slow adoption of modern information and communication technology in health care include the perceptions that it is too expensive, very unusable, and quite divergent from current practice (Schoen et al. 2000). HIT implementations have altered the concerns of health care staff. Now that they are interacting with increased HIT systems in the workplace they are now concerned with the flexibility of the systems, if the systems were "fit for purpose," and have varying levels of confidence and experience with information technology (IT) (Ward et al. 2008). The literature further indicates that proper education and training are required to achieve user acceptance of HIT systems.

In addition, the current generation of HIT systems do not typically come "ready-to-use" off the shelf. It is extremely difficult, if not impossible, for health care provider organizations to have the range of expertise, in house, that is needed to design, adapt, and implement technologies to meet an organization's needs. In fact, many health care organizations are often not even fully aware of their own needs, do not know which technologies are available for what, and do not know how modern information and communication technologies can be effectively used to improve and simplify care delivery. In many cases, IT adoption among health care providers has been driven by Federal and state regulations and requirements rather than well-recognized internal needs and growth opportunities—for example, in Subtitle A—promotion of HIT, the ARRA sets the goal for the utilization of an EHR for each person in the United States by 2014 (American Recovery and Reinvestment Act 2009).

Health informatics is a field that can be widely defined as the generation, development, application, and testing of information and communication principles, techniques, theories, and technologies to improve health care delivery. It includes the understanding of data, information, and knowledge used in the delivery of health care and an understanding of how these data are captured, stored, accessed, retrieved, displayed, interpreted, used, and made more efficient. Although health informatics intersects with the fields of clinical-, biomedical-, medical-, and bio-informatics, it is different in the sense that it focuses on health care delivery, and hence is centered on the patient (and/or consumer), the clinician (or health care professional, or "provider"), and, more importantly, the patient-provider interaction. Human-computer interaction (HCI), from the perspective of both the patient/consumer and the provider, is essential to the success of "health informatics." In this chapter, we first review the characteristics of the healthcare industry in the United States. We then review information systems used by consumers, patients, and providers and raise HCI issues and challenges associated with both perspectives. Then we propose a framework for evaluating health care applications and conclude with a discussion of future opportunities and challenges for HCI in health care.

30.2 CHARACTERISTICS OF THE HEALTH CARE INDUSTRY

30.2.1 Health Care Industry

As previously discussed, the Institute of Medicine (2001) put forth four key underlying reasons for inadequate quality of care in the U.S. health care system today: (1) the growing complexity of science and technology, (2) the increase in chronic conditions, (3) a poorly organized delivery system, and (4) constraints on exploiting the revolution in information technology. In addition, a growing trend toward consumerism has become a major force in shaping the future organization of the health care industry. These five trends, detailed further in the following discussion, continue to shape the future of health care in the United States.

30.2.1.1 Complexity of Science and Technology

The sheer volume of new health care science and technologies—the knowledge, skills, interventions, treatments, drugs, and devices—is very large today and has advanced much more rapidly that our ability to use and deliver them in a safe, effective, and efficient way. Government and private investment in pharmaceutical, medical, and biomedical research and development has increased steadily. The health care delivery system has not kept up with phenomenal advancement in science and technology and proliferation of knowledge, treatments, drugs, and devices. With current advances in genomics (offering promise in diagnosis and, possibly, treatment), sensor technologies (offering promise in automated detection, measurement, and monitoring), nanotechnologies (offering promise in diagnosis, treatment, and control), and information and communication technologies (enabling remote delivery, telemedicine, e-health, and patient empowerment), the complexity of science and technology in health care is only going to increase.

30.2.1.2 Chronic Conditions

As noted by the Institute of Medicine (2001), "because of changing mortality patterns, those age 65 and over constitute an increasingly large number and proportion of the U.S. population" (p. 26). Therefore, there is an increase in both the incidence and prevalence of chronic conditions (defined as conditions lasting more than 3 months and not self-limiting). The National Committee for Quality Assurance (NCQA) found that chronic diseases lead to an estimated 45 million sick days and $7.4 billion in lost productivity each year (2007).

In a landmark study, Hoffman, Rice, and Sung (1996) estimated that American patients with chronic conditions make up 80% of all hospital bed days, 83% of prescription drug use, and 55% of emergency room visits. The presence of chronic disease has become the principal health burden in many developing countries as well. Chronic diseases were responsible for 50% of the disease burden in 23 developing countries in 2005 and will cost those countries $84 billion by 2015 if nothing is done to slow their growth (Nugent 2008). Compared with acute illnesses, effectively treating chronic conditions requires disease management and control over long periods of time, collaborative processes between providers and patient, and patient involvement, self-management, and empowerment.

30.2.1.3 Organization of the Delivery System

The health care delivery system in the United States is a highly complex system that is nonlinear, dynamic, and uncertain. The system is further complicated by a large number of agents who are multiple stakeholders, each with multiple, sometimes conflicting, goals, aspirations, and objectives. As a result, the entire system leads to a lack of accountability; it is as frequently misaligned reward as well as incentive structures, and it suffers from inefficiencies embedded in multiple layers of processes. The health care "product" or "service" is often ill defined or difficult to define and evaluate. The processes involved in delivering health care services are complex, ill specified, and difficult to measure, monitor, and control. Health outcomes are difficult to measure, manage, and analyze. The system experiences growing cost pressures, faces potential insurance premium increases, and is extremely fragmented. Wagner, Austin, and Von Korff (1996) identified five elements needed to improve patients' outcomes in a population increasingly afflicted by chronic conditions as follows:

- Evidence-based, planned care
- Reorganization of practices to meet the needs of patients who require more time and/or resources, and closer follow-up
- Systematic attention to patients' need for information and behavioral change
- Ready access to necessary clinical knowledge and expertise
- Supportive information systems

Regarding this last point, the Institute of Medicine (2001) pointed to the fact that

> Healthcare organizations are only beginning to apply information technology to manage and improve patient care. A great deal of medical information is stored on paper. Communication among clinicians and with patients does not generally make use of the Internet or other contemporary information technology. Hospitals and physician groups operate independently of one another, often providing care without the benefit of complete information on the patient's condition or medical history, services provided in other settings, or medications prescribed by other providers. (p. 30)

In the 10 years since that IOM reporting, progress has been made in applying information technology to manage and improve patient care, but widespread adoption across the industry has still not been achieved. Thus, fragmentation of patient information and gaps in care coordination across providers treating the same patient continue to be a significant problem.

30.2.1.4 Information Technology

The revolution in information technology holds great promise in a number of areas for consumers, patients, clinicians, and all organizations involved in the delivery of health care services. A report by the National Research Council of The National Academies (2000) identified six major information technology applications domains in health care: (1) consumer health, (2) clinical care, (3) administrative and financial transactions, (4) public health, (5) professional education, and (6) research. There have been notable technological advances in health care. For example, in the second edition of this handbook, it was noted that while some applications were currently in use (such as online searches for medical information by patients), many other applications were in early stages of development (such as telerobotic remote surgery). Since then, the world's first telerobotic remote surgery service was established (Anvari, McKinley, and Stein 2005), and surgical teleproctoring is an accepted technique for guiding remote general surgeons by a surgical subspecialist when patients are in need of an emergency subspecialty operation (Ereso et al. 2010).

Although the Internet (and intranets) has been a driving force for changes in the information technology landscape in the past decade, many health care applications are not web based. Applications such as administrative billing systems, EHRs, and computerized physician order entry (CPOE) systems frequently remain on legacy systems, and lack integration with other applications. The current generation of EHR and CPOE systems had begun to provide web access to these systems; this access is sometimes provided via third-party applications like Citrix, which introduce usability issues related to single-sign-on access and performance.

With respect to consumers/patients and providers, the Committee on the Quality of Health Care in America identified five key areas in which information technology could contribute to an improved delivery system (Institute of Medicine 2001):

- Access to medical knowledge base
- Computer-aided decision-support systems (DSSs)
- Collection and sharing of clinical information
- Reduction in medical errors
- Enhanced patient and clinician communication

The ensuing and ongoing infusion of information technology into the delivery of health care has yet to fully achieve success in these five key areas, despite the fact that it has been 10 years since they were first identified. One factor contributing to delays in adoption is poor usability of these systems and the disruption they cause to clinical workflows. Exacerbating this problem is a limited supply of IT personnel knowledgeable in both IT and clinical workflows and challenges who can help to identify and address these usability and workflow issues to avoid potential safety issues, improve efficiency of IT use, and ultimately increase technology acceptance by clinicians.

30.2.1.5 Consumerism

Consumerism in health care can be defined as an "orientation to new care delivery models that encourage and enable greater patient responsibility through the intelligent use of information technology" (Cohen et al. 2010, p. SP37). The Internet and other developments in information and communication technologies are contributing to greater consumerism with stronger demands from individuals for information and convenience. People are more demanding; they want timely and easy access to medical information, the latest in technology, and the latest in customer service. However, getting access to one's own health data in a format that is easy to understand is nearly impossible, even for those who have some knowledge about health matters, are well educated, and do not have physical, perceptual, or cognitive disabilities (Jacko 2011). Researchers and providers are beginning to recognize that patients need access to tools that can lead to empowerment and shared decision making regarding their own health care. In addition, because of increased fragmentation and specialization in medical care, patients need to take a more active role in managing their health and health care to ensure that the various providers involved in providing their care (i.e., primary-care physician, specialist, pharmacist, etc.) have the information they need to provide appropriate, quality care. Cohen et al. (2010, p. SP37) state that although consumerism holds promise, the following four principles should be followed to deliver improved outcomes and lower costs: "(1) keep the consumer at the center of innovation, (2) keep it simple, (3) link products and services to a broader 'ecosystem' of care, and (4) encourage health in addition to treating illness."

However, there are many technical, organizational, behavioral, and social challenges and barriers to an increased use of information technology by individuals in managing their own health/health care. These challenges not only include the design of optimal, effective, flexible human–computer interfaces but also issues of privacy/security of information, health literacy, and challenges associated with health information exchange (HIE). Those are discussed in Section 30.2.2.

30.2.2 Health Care Regulatory Environment in the United States

The health care industry has experienced unique legislative milestones (Kumar and Chandra 2001). Among them, three in particular have had far-reaching implications for the field of health informatics: (1) the Health Insurance Portability and Accountability Act (HIPAA); (2) health information and other business data security; and (3) The Health Information Technology for Economic and Clinical Health Act (HITECH Act). The HIPAA Act was passed and signed into law in 1996 and is designed to improve the portability of health insurance coverage in group and individual markets, limit health care fraud and abuse, and simplify the administration of health insurance. The act has serious, impending implications for health care providers and information managers. Of all its mandates, administrative simplification is perhaps the most critical for health care information managers, who are faced with everything from establishing standardized financial and clinical electronic data interchange (EDI) code sets to adopting, assigning, and using unique numerical identifiers for each health care provider, payer, patient, and employer. Both HIPAA and the growing role of the Internet-based technologies in delivering health care create an even greater and critical concern for data security and privacy.

The HITECH Act, enacted as part of the ARRA of 2009, provides an unprecedented investment in HIT to improve American health care delivery and patient care. The provisions of the HITECH Act are

> specifically designed to work together to provide the necessary assistance and technical support to providers, enable coordination and alignment within and among states, establish connectivity to the public health community in case of emergencies, and assure the workforce is properly trained and equipped to be meaningful users of EHRs. Combined, these programs build the foundation for every American to benefit from an EHR, as part of a modernized, interconnected, and vastly improved system of care delivery. (Office of the National Coordinator for Health Information Technology, 2011)

The interconnected programs that were funded by the HITECH Act are as follows:

- Beacon Community Program
- State HIE Cooperative Agreement Program
- HIE Challenge Grant Program
- HIT Extension Program
- Strategic Health IT Advanced Research Projects Program
- Community College Consortia to Educate HIT Professionals Program
- Curriculum Development Centers Program
- Program of Assistance for University-Based Training
- Competency Examination for Individuals Completing Non-Degree Training Program

Under the HITECH Act, privacy and security of personal health data (PHI) remain of paramount concern. Covered entities are required to report any breach of health information affecting 500 or more individuals to the Secretary of the Department of Health and Human Services (HHS) within 60 days of discovery. These breaches are posted on the HHS website for consumers to access. As of March 14, 2011, a total of 241 breaches have been listed since September 22, 2009 affecting a total of 8,225,193 individuals (United States Department of Health and Human Services 2011).

30.2.3 Meaningful Use of Electronic Health Records

The HITECH portion of ARRA specifically mandated that incentives should be given to Medicare and Medicaid providers, not for EHR adoption, but for "meaningful use" (MU) of EHRs. This distinction is important from the perspective of HCI because "meaningful use" is nearly impossible without highly usable technologies and systems. In July 2010, the U.S. Department of HHS released that program's final rule, thus defining stage 1 MU. Furthermore, that the proposed draft criteria for stages 2 and 3 of MU aggressively raise the requirements for EHR use to improve advanced care processes and health outcomes (United States Department of Health and Human Services 2011). The MU objectives and measures focus on five health outcome priorities: (1) improving quality, safety, efficiency, and reducing health disparities; (2) engaging patients and families in their own care; (3) improving care coordination; (4) improving population and public health; and (5) ensuring adequate privacy and security protections for personal health information.

Many of the criteria focus on clinician utilization of specific EHR functionality, thus HCI considerations that impact clinician acceptance and use of EHRs are critical to achieving the MU program goals. Also, the need for optimizing the interaction between patients/caregivers/families and (health) information technology is clear in the MU objectives and measures, particularly in the objectives and measures relating to engaging patients and families in their care. Notably, the evidence base and rationale provided for the proposed new MU objectives cite landmark studies and clinical trials (e.g., Gustafson et al. 1999; Mark et al. 2008; Ralston et al. 2007; Riggio et al. 2009; Rosenberg et al. 2008; Gustafson et al. 2001) that demonstrate the inherent value of providing patients and their families with highly usable tools and technologies that enable them to actively engage in their own health and health care.

30.2.3.1 Transaction Standards and Coding Sets

The U.S. Department of HHS has adopted national standards for electronic administrative and financial healthcare transactions. This is one of the most positive attributes of Title II of HIPAA, also known as administrative simplification, as it eliminates the conflicting transaction standards, coding sets, and identifiers used by the various players in the industry. Initial transaction and code sets were implemented in October 2002, with a revised code set standards defined and implemented by October 2003. The standards relating to enrollment, referrals, claims, payments, eligibility for a health plan, payment and remittance advice, premium payments, health-claim status, and referral certifications and authorizations. By developing national standards, it is anticipated that EDI of healthcare data will significantly reduce administrative costs. The Workgroup on Electronic Data Interchange (2000) estimated that EDI can reduce administrative costs by $26 billion per year by streamlining precertification, enrollment status, and reimbursement processes. Code Sets have also been defined in which variations are not permitted. Code Sets are based on the following standards:

International Classification of Diseases, Ninth Revision (ICP-9) Volumes 1 and 2—Diagnosis Coding
International Classification of Diseases, Ninth Revision (ICP-9) Volume 3—Inpatient Hospital Service Coding

International Classification of Diseases, Tenth Revision, Clinical Modification (ICD-10-CM) (To be implemented October 1, 2013)

International Classification of Diseases, Tenth Revision, Procedural Coding System (ICD-10-PCS) (It is a successor to Vol. 3 of ICD-9-CM as well as a clinical modification of the ICD-10) (To be implemented October 1, 2013)

Current Procedural Terminology (CPT 4; Fourth Edition, maintained by the American Medical Association [AMA].)—Physician Service Coding

Current Dental Terminology (CDT; it is a code set developed and maintained by the American Dental Association [ADA].) 2011/2012—Dental Service Coding

Healthcare Common Procedure Coding System (HCPCS; based on CPT. It is pronounced "hick picks.")—Other Health-Related Coding

DRG—Diagnosis Related Groups

NDC—National Drug Coding

HIPAA also establishes unique identifiers for health care providers, health plans and payers, employers, and eventually patients/individuals. Currently, the patient/individual identifier is pending as no consensus could be reached.

30.2.3.2 Privacy and Security

Administrative simplification also addresses patient privacy and security. Privacy standards exist for disclosure of patient identifiable information (including demographic data), training of healthcare workforce, individual's rights to see records, procedures for amending inaccuracies in medical records, maintenance of privacy when patient information is exchanged among business associates, designation of a privacy officer, procedures for complaints, sanctions for infractions, duty to mitigate, and document compliance. As a component of disclosure, covered entities (providers, payers, and clearinghouses) must make reasonable efforts not to use or disclose more than the minimum amount of protected patient information necessary to accomplish the intended purpose of the use of disclosure. Wide adoption and use of electronic medical record (EMR) systems will present technical challenges to preventing disclosure of this information. In the past, accessibility of paper-based patient records to clinicians was restricted in part by the chart's physical location (i.e., in medical records storage, on the nursing unit in a hospital). However, EMR makes these charts available to a much large number of clinicians in a wider range of physical locations, making it more difficult to prevent inappropriate uses of this protected information by authorized users of the EMR.

While the privacy rule of HIPAA pertains to all Protected health information (PHI), the security rule deals specifically with electronic protected health information. Security standards require establishment of administrative procedures (policies and procedures), physical safeguards (physical access to computers), technical security (individual and network computer access), and electronic signature (optional, but if used, it must be digital). Implementation requirements of the privacy and security standards have received a tremendous amount of debate. Although most feel the intent of the standards is good, the cost implications have left many advocating for a longer implementation period. The Department of HHS previously estimated the total cost to implement Subtitle II to be approximately \$17–\$18 billion over the next 10 years. The Mayo Clinic collected startup cost data for HIPAA implementation prospectively from January 1, 2001, through April 14, 2003. Their costs included direct (policy revisions) and indirect costs, training, forms revision and printing, privacy curriculum, workforce training, privacy expenses and IT costs, totaling \$4,663,672 (Williams et al. 2008).

In addition to the cost of implementation, the privacy and security requirements present usability challenges as well, for example, security steps required for patient access to a patient portal can be time-consuming and cause some patients not to register and use these online tools. Additionally, striking the right balance between protecting patient information but ensuring that clinicians can access it in emergency medical situations continues to be debated in both large provider organization and state and community HIEs. Requirements for auditing access to PHI also have significant IT system performance implications, and those performance issues can translate to user dissatisfaction. Thus, more attention to usability issues related to privacy and security requirements is needed.

30.3 HEALTH INFORMATICS

As defined earlier, health informatics comprises the generation, development, application, and testing of information and communication principles, techniques, theories, and technologies to improve the delivery of health care with a focus on the patient/consumer, the provider, and, more importantly, the patient–provider interaction. Although a number of systems using a number of platforms and technologies have already been developed and are currently being developed, the field itself is still in its infancy. In addition, current systems have been designed to fit within the existing health care delivery system and thus are only marginally or superficially impacting the way health care is being delivered. The true potential of health informatics is yet to be experienced and will radically transform the way health care is being delivered and managed in the future. Sections 30.3.1–30.3.6.3 provide background information on current health care applications with an emphasis on two types of users: (1) the consumer/patient and (2) the clinician/provider.

30.3.1 Informatics from a Consumer-Centered Perspective

Consumer health information has been defined as "any information that enables individuals to understand their health and make health-related decisions for themselves or their families" (Harris 1995, p. 210). Patrick and Koss (1995) listed a variety of organizations and entities that produce and/or disseminate consumer health information including health-related organizations (involved in provision of, or payment for, health care services and supporting services), libraries, health voluntary organizations (i.e., American Heart Association, American Cancer, American Lung, etc., and 60,001 other health interest societies), broadcast and print media, employers, government agencies, community-based organizations (i.e., churches, YMCA, agencies for the elderly), networked computer health information providers—"virtual" communities. Methods of dissemination are diverse and include informal channels, printed text, broadcast electronic media, dial-up services (telephone), nonnetworked computer-based information (i.e., CD-ROM, Kiosk technology), and networked interactive computer-based information.

The field of Consumer Health Informatics (CHI) is focused on supporting the consumer needs for health information. The goal of CHI is to develop web-based applications for consumers to be able to efficiently obtain and understand information found online. CHI also analyzes individual needs for health information, accessibility of information, and the integration of consumer preferences into online applications (Eysenbach et al. 2005). CHI applications range from smoking cessation to asthma/COPD websites. Select CHI applications have been found to successfully engage consumers, enrich traditional clinical interventions, and improve health outcomes (Gibbons et al. 2009).

Harris (1995) discussed a number of problems with health information including how to interpret conflicting or differing information; how to judge reliability; how to choose among many alternatives; and how to deal with the vast quantities of information, much of which is superficial or even inaccurate. Over 80% of Americans have searched for health information online, yet only 15% of those say they "always" check the source and date of the information they find (Fox 2006). This indicates that the majority of information seekers online are not paying attention to quality indicators that are available.

30.3.2 Interactive Health Communication

Historically, print and broadcast dissemination of health information has led to a number of problems such as the timing relative to need, single directionality (difficulty of following up, clarifying, and understanding), timeliness and relevance vis-à-vis updates, and uniqueness to individuals (Patrick and Koss 1995). As early as 1995, Harris asserted that electronic sources of health information had the potential to be more timely and complete as other media and could become more accessible to all citizens. Consequently, the use of interactive health communication or CHI has emerged as a dominant theme.

Web 2.0 applications are becoming increasingly popular for clinicians, students, and health care consumers. Hughes et al. (2009) found that despite concerns over validity of information, over half of the physicians they followed utilized Web 2.0 applications. Wikis, blogs, and podcasts are most popular because of their ease of use, collaboration, and rapid deployment of information sharing (Boulos, Maramba, and Wheeler 2006). Unfortunately, wikis can be edited by anyone, which means that the information provided to the consumer is not always reliable and accuracy is difficult to assess. Blogs and podcasts are created by individuals, and therefore somewhat easier to gauge for accuracy depending on the source. The Cleveland Clinic, and any other health systems, has libraries of reliable health and wellness videos available as vodcasts, or video podcasts (e.g., http://my.clevelandclinic.org/health_edge/vodcasts.aspx). Topics range from tips on how to change your diet to the difference between a cold and sinusitis. Although some initial applications of Web 2.0 technology are promising, more research is needed to ensure security and reliability of information for health care consumers.

30.3.3 Consumer/Patient Web-Based Applications and e-Health

An estimated 122 million Americans seek health information online (Tu and Cohen 2008). In 1999, the Benton Foundation (1999) estimated there were 10,000 or more health-related websites. However, this number has grown tremendously to meet increasing consumer demand for health care information. In early 2011, the Google directory of health-related websites (http://www.google.com/Top/Health/) listed over 19,000 websites related to diseases and conditions alone and thousands of other sites addressing topics ranging from pharmacy to alternative medicine to public health. This variety of information available demonstrates that consumers and patients desire a range of information and services including getting disease treatment information, obtaining report cards on physicians or hospitals, exchanging information with other patients, interacting online with their physicians, and managing their own health benefits. The sheer volume of information on the Internet exceeds most expectations yet raises problems: efficient search for information, information retrieval, information visualization, human-information processing, understanding, and assimilation. Authors of Internet information need to determine how to best structure the information for use by others. E-Health can be described as the transition of health care processes and transactions into the Internet-delivered electronic superhighway. Potential problems exist that are specific to patients seeking electronic health-related information (Sonnenberg 1997), including the lack of editorial control of information, conflict of interest for website sponsors, and unfiltered information presenting an unbalanced view of medical issues. However, referring primarily to self-help groups, Ferguson (1997) believed that the problem of obtaining bad medical information online is not very different from obtaining bad medical information at cocktail parties, in the tabloids, in magazine ads, and so on. In addition to the content of information, there are technological problems that may affect both access and use of health information on the Internet: slow modems, poor institutional Internet connections, firewalls that interfere with Internet traffic, malfunctioning message routers, and heavy Internet usage in the immediate geographic area (Lindberg and Humphreys 1998). Most current concerns about interactive health communication center on the fact that these applications have the potential to both improve health and to cause harm; thus, highlighting the need to ensure their accuracy, quality, safety, and effectiveness (Robinson et al. 1998).

30.3.4 Personal Health Records

The personal health record (PHR) is an example of an interactive technology that is designed for consumer engagement. The U.S. health care system can be fragmented and often creates passive patients with high-cost dependency on the health care system. The use of PHRs can change this by transforming the consumer's role while improving the delivery of health care and also facilitating research. The Center for Information Technology Leadership (CITL), a nonprofit research center, released findings on the value of PHRs, concluding that PHRs could save $19 billion annually on a national level by focusing on prevention, early intervention, self-management, and evidence-based care (Kaelber et al. 2008). CITL reported that PHRs, web-based systems that allow patients to populate and maintain their medical data, continuity of care documents, prescription medication lists, health histories, hospital discharge summaries, can also enable consumers to elect to share their information with third parties to facilitate wellness. PHRs empower patients to use their data to better manage their health care and health care costs. The nonprofit Partners Center for Connected Health released a report concluding that EMRs will not reform health care without consumer engagement, stating: "True health care reform will require a more patient-centered approach and a broader policy palette, including incentives for providers to adopt more population health management tools and for patients and consumers to take more ownership of their health" (Kvedar 2008). The goal is for PHRs to serve as integrated tools for health management (Masterson 2008). Some health care organizations have begun offering PHRs to their patients at no cost. Through the PHR, patients are provided access to a portal that enables, for example, viewing and archiving laboratory results, scheduling appointments, and securely messaging their physicians.

30.3.5 Health Informatics from a Provider-Centered Perspective

Providers and health care provider organizations have long used health-information systems to support both administrative and clinical functions of health care delivery and management. However, despite the fact that health care is one of the most information-intensive industries; it has very few state-of-the-art information management systems. Health care is fragmented, with hundreds of thousands of payers, hospitals, physicians, laboratories, medical centers, pharmacies, and clinics, each with its own legacy of systems, hardware, software, and platforms. EDI and connectivity issues have become critical. Numerous information systems have been developed and implemented, the most noteworthy being integrated EMRs/EHRs and CPOE systems. The following is a discussion of some important clinical applications.

30.3.5.1 Electronic Medical Records

EMRs are increasingly being adopted in both primary-care and inpatient-care environments. EMRs provide functions to document all clinical processes and patient-related information relevant to the delivery of patient care. EMRs are advocated and used to improve the quality, accessibility, and timeliness of patient medical information. However, because EMRs replace traditional patient charts, they affect a wide range of clinical users and clinical processes. As such, they face a number of barriers to adoption, including cost, a lack of tested systems, problems with data entry, inexperienced vendors, confidentiality concerns, and security concerns (Wager et al. 2000). In a systematic literature review of articles published between 1998 and 2009, eight categories of barriers to EMR adoption were identified. Those categories are: (A) Financial, (B) Technical, (C) Time, (D) Psychological, (E) Social, (F) Legal, (G) Organizational, and (H) Change Process (Boonstra and Broekhuis 2010). Understanding barriers to implementation and successful interventions should allow an increasing number of organizations to implement EMRs.

In addition, EMRs may have barriers that are specifically related to the practice of medicine. For example, physician use of an EMR while with patients could affect patient perceptions of quality of care or quality of physician–patient interactions. Although some past studies examining this issue have not shown this to be the case (i.e., Legler and Oates 1993), a survey of pediatric-care primary physicians using EMR indicated that they felt it reduced eye contact with patients and increased the duration of patient visits (Adams, Mann, and Baushner 2003). Thus, the impact of these technologies on physician-patient interactions is not clear at this point.

A key factor that will contribute to whether EMR systems become more rapidly adopted in primary-care practices is whether or not physicians themselves perceive the systems as improving quality (of the medical records, patient care, and overall performance). Anderson and Balas (2006) surveyed U.S. primary-care physicians and found that nearly 75% indicated that HIT could reduce errors; 70% felt HIT could increase their productivity; over 60% indicated that HIT could potentially reduce costs and help patients assume more responsibility.

Another study examined this issue using qualitative methods in five community-based practices that had used EMR for at least 2 years and did not use a duplicate record system (Wager et al. 2000). Results indicated that many physicians and staff members perceived benefits, such as increased access, an ability to search the system, and improved overall quality of medical records. However, there were several disadvantages mentioned, including the frequency of downtime and the time necessary to develop customized templates. A more recent survey found additional concerns such as physician over-reliance on potentially inaccurate information, nurses focusing more on EMR use protocol than reviewing order accuracy, neglected orders in the system, and physicians ignoring alerts because they were used to false alarms (Holden 2010).

In a study examining the quality of worklife of family physicians, Karsh et al. (2001) collected quantitative data about EMR to compare perceptions of medical records between physicians using EMR and those not using EMR. Specifically, they assessed whether or not the use of EMR was related to perceptions of improved quality of medical records. The results showed that physicians who used EMR perceived their medical records to be more up to date and accessible. Physicians who used EMR were also more satisfied with the overall quality of their medical records systems. However, there were no differences in perceptions of whether medical records could be modified to meet individual needs. This suggests that EMR can have positive impacts on medical records. (It is possible, though, that the positive responses were caused not by positive traits of the EMR, but rather because of cognitive dissonance.)

The results of Karsh et al. (2001) study supported those of qualitative studies of physicians who use EMR. Wager et al. (2000) found that physicians in primary practice who had used EMR for at least 2 years believed EMR to have many benefits over paper-based systems, including increased access and availability of patient information to multiple users, the ability to search the system, improved overall quality of patient records, improved quality of documentation, increased efficiency, facilitated cross-training, and improved communication within the practice. Thus, it is clear that EMR has the potential to improve the quality of patient records and therefore, possibly, the quality of care. However, to fully capitalize on such systems, they need to be designed to maximize usability, connectivity, and portability while guaranteeing privacy and security.

30.3.5.2 Electronic Health Records

Although the terms EMR and EHR are often used interchangeably in the market and by government officials, they are not equivalent terms. The National Health Alliance for Information Technology (2008) penned the following definitions to describe the differences: "EMR: An electronic record of health-related information on an individual that can be created, gathered, managed, and consulted by authorized clinicians and staff within one health care organization. EHR: An electronic record of health-related information on an individual that conforms to nationally recognized interoperability standards and that can be created, managed, and consulted by authorized clinicians and staff across more than one health care organization."

EHRs are currently in their infancy because they are dependent on the ability to exchange health information through a Regional Health Information Organization, state or community HIE or through the proposed National Health Information Network. The ability for EHRs to be implemented and integrated across communities or regions will be reliant on the establishment of clinical information transaction standards for EMR. The investment in HITECH, for example, the Beacon Program, gets us closer to achieving this goal.

Underlying all of this is the ability and willingness of providers to actually use the system. Edwards et al. (2008) evaluated a pediatric EHR system using a predictive evaluation method prior to EHR implementation at a major pediatric health system in the Southeastern United States. The method resulted in many system improvements and enabled avoiding usability problems at time of system rollout. This case study demonstrated both the need to conduct usability evaluations for large commercial software implementations and the utility of using heuristic walkthrough as a tool for accomplishing this.

30.3.6 Health Information Exchange

An HIE is used to "electronically capture and distribute administrative and clinical information between health care stakeholders while maintaining the meaning of the information exchanged via a common language" (Krohn 2008, p. 7). With the incentives and funding provided in the ARRA act to promote HIE adoption and use, implementing HIEs is receiving increased attention. Some of the benefits that can be achieved through HIE include the following:

- Improved care coordination at transitions in care
- Improved patient safety
- Improved communication among providers caring for the same patient
- Increased hospital and ambulatory provider efficiency and reducing administrative costs
- Improved care quality
- More effective use of health care resources (e.g., reducing duplicate laboratory tests)
- Improved population health

However, early HIEs have struggled due to a number of challenges. These challenges include financial sustainability, physician adoption, privacy/security concerns, data sharing policies, lack of technical standards, and workflow challenges. The e-Health Initiative has identified 234 active HIEs in the United States, 199 of which responded to the annual 2010 Survey of HIE. The majority of respondents reported a reduction in staff time and redundant testing, and increases in functionality with respect to the MU rules (e-Health Initiative 2010). In addition, 44 of these initiatives now allow patients to view their health data, which is up from 3 in 2009.

Current HIEs are working through these challenges, but clear best practices have yet to emerge. With HIEs, achieving the benefits is directly related to the degree of adoption and data sharing across the community, so HCI methods need to be applied to maximize the use and, consequently, benefit of HIEs.

30.3.6.1 Computerized Physician Order Entry Systems

CPOE systems have substantial potential for improving the medication ordering process because they enable physicians to write orders directly online. They have the potential to ensure complete, unambiguous, and legible orders; they assist physicians at the time of ordering by suggesting appropriate doses and frequencies, by displaying relevant data to assist in prescription decisions, by checking drugs prescribed for allergies, and drug–drug interactions. As such, those systems are believed to potentially reduce the incidence of medical errors in general, and medication errors in particular. Research has demonstrated that CPOE systems can reduce medication errors (Bates et al. 1999; Kuperman et al. 2007; Kadmon et al. 2009). In addition, the use of CPOE can decrease turn-around time for medication orders as well as radiation and laboratory orders (Mekhjiane et al. 2002) and improve adherence to clinical guidelines (Overhage et al. 1997). A survey of ambulatory care providers assessed attitudes toward CPOE and e-prescribing systems and found that the majority of such providers reported improved quality of care and efficiency, prevention of medical errors, and increased patient satisfaction as advantages to the system. Despite this, only 47% of providers reported satisfaction with the system. Providers expressed dissatisfaction with alert fatigue, and alerts inappropriately identifying drug interactions (Weingart et al. 2009).

Despite the demonstrated advantages, in 2007 only an estimated 12.0% of hospitals were using CPOE with decision support, and less than a quarter were using bar-code medication administration (Pedersen and Gumpper 2008). One may conclude that CPOE systems are not a panacea. In some cases, their use has resulted in negative outcomes, including increased physician order entry time (Bates, Boyle, and Teich 1994; Shu et al. 2001) and an increased coordination load on clinical care teams (Cheng et al. 2003). Physicians have also reported that CPOE adds additional burden on the physician because it requires many clicks and screen navigations instead of simply writing an order (Holden 2010).

For computerized physician-order entry systems to be fully successful, physicians need to use them. However, as the studies previously discussed indicate the gains achieved come at a cost, frequently imposed on the physicians using the system. This leads to issues of HCI design, usability, and integration within the care delivery processes to ensure that the burden on these users is minimized. The importance of addressing these issues has also been highlighted in several recent studies showing that the potential for new types of prescribing errors can be created by usability problems in the interface and/or lack of integration of the system with clinical work practices (Koppel et al. 2005). In certain care situations, such as critical-care environments, this lack of integration between the clinical processes and patient care needs (i.e., urgency of care) can have dire consequences for patients (Han et al. 2005). This emphasizes the importance of identifying and resolving potential usability problems prior to implementation, continually monitoring their use, and incrementally improving their usability in the context of the needs of specific clinical care environments.

30.3.6.2 Patient Monitoring Systems

Gardner and Shabot (2001) defined patient monitoring as “repeated or continuous observations or measurement of the patient, his or her physiological function, and the function of life support equipment, for the purpose of guiding management decisions” (p. 444). Electronic patient monitors are used to collect, display, store, and interpret physiological data. Increasingly, such data are collected using newly developed sensors from patients in all care settings and in patients’ own homes. Although such data can be extremely useful for diagnosis, monitoring, alerts, as well as treatment suggestions, the amount, diversity, and complexity of data collected presents challenges to HCI design. Although advancements have been made in telemonitoring, conclusions from research studies vary immensely. A recent Cochrane review article found that telemonitoring of heart failure patients reduced the rate of death by 44% and the rate of hospitalizations related to heart failure by 21% (Inglis et al. 2010). In a larger study, Chaudry et al. (2010) found that telemonitoring for heart failure patients did not improve outcomes and stress the importance of a comprehensive, independent evaluation of disease management strategies prior to adoption.

30.3.6.3 Imaging Systems

Imaging is a central element of the health care process for diagnosis, treatment plan design, image-guided treatment, and treatment-response evaluation. Greenes and Brinkley (2001) noted that the proliferation in number and kind of images generated in health care led to the creation of a subdiscipline of medical informatics called “imaging informatics.” In their review and summary of the field, Greenes and Brinkley noted that

> as processing power and storage have become less expensive, newer, computationally intensive capabilities have been widely adopted. Widespread access to images and reports will be demanded throughout health care delivery networks.... We will see significant growth in image-guided surgery and advances in image-guided minimally invasive therapy as imaging is integrated in real time with the treatment process. Telesurgery will be feasible. (pp. 534–536)

They also highlight that such ambitious evolution of imaging systems will be in part dependent on continued advances in user interfaces and software functionality.

Picture archiving and communication system (PACS) has revolutionized the practice of radiology over the last 15 years (Huang 2010). These complex systems require knowledgeable radiologists as end-users and a team of HIT professionals to keep the systems running. Large quantities of data pass through any PACS (over 50 GB per day), while radiologists and clinicians demand constant access to the images (Branstetter 2007).

30.4 CLINICAL AND PATIENT DECISION-SUPPORT SYSTEMS

DSS are computer systems used to support complex-decision making. The goal of DSS is to improve the efficiency and/or effectiveness of decision-making processes (Shim et al. 2002). As such, many of the complex-decision processes in health care can benefit from the use of DSS. One class of DSS in health care is clinical decision-support systems (CDSS), which provided decision support to clinical users during the process of providing patient care. Another class of DSS, patient decision-support systems (patient DSS), is designed to educate and support patients as they make decisions that affect their health care. Described here are two types of health care-related DSS.

Musen, Shahar, and Shortliffe (2006) defined a CDSS as "any computer program designed to help health professionals make clinical decisions" (pp. 707–708). They characterized CDSS on five dimensions as follows:

1. System function
2. Mode for giving advice
3. Style of communication
4. Underlying decision-making process
5. HCI

A large number of applications already exists and can be further developed. The opportunities are almost unlimited. Bates et al. (2001) proposed that appropriate increases in the use of information technology in health care, especially the introduction of CDSS and better linkages in and among systems, could result in substantial reduction in medical errors. For example, studies have demonstrated that implementation of CPOE systems that include CDSS functions (i.e., drug interaction checking, allergy checking, etc.) have reduced medication errors and adverse drug events in both general-care settings (e.g., Bates et al. 1999) and pediatric-care settings (i.e., Potts et al. 2004; Cordero et al. 2004).

However, as noted by Musen, Shahar, Shortliffe (2006), "systems can fail … if they require that a practitioner interrupt the normal pattern of patient care" (p. 712). Clinicians often have to perform a manual intervention due to data mismatch, a common issue within the DSS community (Acosta et al. 2010). Thus, new technologies (mobile devices, wireless networks, and distance-communication technologies) and novel HCIs (based on speech, gestures, and virtual reality) offer huge potential to maximize usability and permit seamless integration of CDSS within complex, dynamic work processes.

A roadmap for national action on clinical decision support (CDS) was created in 2005 by the American Medical Informatics Association with assistance from the Office of the National Coordinator and the Agency for Health care Research and Quality. The three pillars for realizing the potential of CDS were identified as: (1) best knowledge available when needed; (2) high adoption and effective use; and (3) continuous improvement of knowledge and CDS methods (Osheroff et al. 2007; Lyman et al. 2010). Great progress has been made toward these goals over the past several years. More health care organizations are utilizing HIT and CDS to improve the quality of patient care. Research articles highlight HIT success with alerting and reminder systems, dosing calculators, and order sets. As many of the strategic objectives of the 2005 roadmap have been met, it is important to redefine the goals and objectives to continue making progress in CDS.

The medical industry has increasingly acknowledged the need to enable patients to participate in making health-related decisions. To effectively participate in decision making, most patients need to become better informed about the options available and need help assimilating that information to apply it to the decision at hand. These health-related decisions range from choosing a health-insurance plan to working with their doctor to select a treatment for cancer.

Patient DSS are one of several types of decision aids used to help patients participate in health-related decisions. Decision aids are interventions provided to assist individuals as they make a deliberative choice between two or more alternatives (Bekker, Hewison, and Thornton 2003). Patient DSS supports a patient in one or more stages of making a health-related decision. Much of the past and current patient DSS research and development efforts have targeted patients with life-threatening or chronic diseases. Most have focused on supporting decisions regarding treatment options (i.e. medical or surgical therapies), although a few have examined early detection and other issues (O'Connor 1999). Patient DSS that support patients faced with treatment decisions provide one or more of the following functions that facilitate patient participation in decision making (O'Connor 1999; Scott and Lenert 1998):

Educate the patient. Provide the patient with information about the treatment alternatives and outcomes, especially highlighting risks and benefits associated with each treatment alternative.

Tailor information. Tailor information content and/or presentation based on patient characteristics such as their health and demographics factors.

Assess preferences. Use preference-elicitation methods to assess the patient's values/preferences for the possible intervention outcomes.

Optimize decision. Optimize the decision outcome based on context, heuristics, probabilities of outcomes, and algorithms.

Reviews of patient decision aids including patient DSS (i.e., Molenaar et al. 2002; O'Connor 1999; O'Connor et al. 1999; O'Connor et al. 2003) have revealed that "decision-support strategies have received generally consistent positive ratings by patients in terms of feasibility, acceptability, length, balance, clarity, amount of information, and usefulness in decision making" (O'Connor 1999, p. 260). Additionally, they demonstrate a number of benefits for patients, including improved decision-relevant knowledge, reduced decisional conflict, improved congruence between values and choice, more realistic expectations, more active patient participation, and fewer patients who are unable to make a decision. Bekker, Hewison, and Thornton (2003) proposed that many observed benefits are a result of decision aids enabling patients to use more effective cognitive and emotional strategies. Note that the reviews do not indicate a consistent, significant effect of decision aids on decision satisfaction, although a generally positive impact on satisfaction with the decision process is indicated. Use of decision aids tends to reduce the number of patients who take a passive role in decision making and the number who are undecided (O'Connor et al. 2003).

Patient comprehension of medical terminology is an issue for patient DSS. Patient vocabularies and knowledge of medical terminology have been demonstrated to affect comprehension (Keselman et al. 2007). To improve patient understanding, research is now being conducted to assist patients in understanding clinical information such as laboratory reports and discharge summaries. Successful techniques have included using prototype translators (Zeng-Treitler et al. 2007) and supplementing online discharge summaries with automatically generated hyperlinks and embedded readability support (Adnan, Warren, and Orr 2009; Adnan et al. 2010). As patient DSS continue to advance, it is important to ensure that the information the patients are accessing is comprehensible.

In addition to being used to support patients faced with treatment choices, similar support systems can be used to help healthy individuals make health-related choices. For example, DSS can support health care consumers as they make the important choice of choosing health insurance. Considerable effort has been made to educate consumers about health insurance. One system available for learning about health insurance plans is the health plan report card created by the NCQA (The National Committee for Quality Assurance, 2011). The report card is, in effect, an interactive decision-support tool that helps consumers compare and select a health plan. The report card provides information on a variety of attributes for each plan and enables consumers to identify plans that meet their specifications. This system provides access to plan quality information and uses an easily comparable matrix format and a "star" rating system to simplify comparisons of plans. Despite the increasing availability of information on health plans, many consumers still find the information confusing and the decision difficult. By presenting health-plan information in a way that supports and, perhaps, adapts to a consumer's characteristics and their health-plan preferences, we may improve the usability of the information and enable consumers to make more informed decisions about their health insurance coverage.

30.4.1 Patient Preferences

Patients' preferences and priorities regarding their health and health care are applied each time they participate in making health-related choices. However, an individual's stated preferences can be influenced by external factors. For example, researchers have found that framing of information about treatments is handled (positive, negative, or neutral) has affected patients' treatment preferences (Llewellyn-Thomas, McGreal, and Thiel 1995). It is likely that these tendencies for external factors to influence individual preferences will hold for a variety of health-related choices. Therefore, presenting decision makers with information previously not considered in their decision may cause them to change their preference structure. Slovic (1995) viewed "preference construction as an active process" (p. 369). Taking this view, it is important for patient DSS designers to take a patient-(user) centered approach to the design of these systems to ensure that patient decision outcomes are improved and to avoid introducing systematic biases into the decision process. By structuring the education and decision process to help the patient develop a comprehensive picture of relevant facets of the decision, we can help the patient develop a more comprehensive, rational set of preferences to apply to his or her health choices. In this way, DSS can help patients/health care consumers make more informed decisions.

30.4.2 Human–Computer Interaction Challenges in Patient Decision-Support Systems

The heterogeneity of patient/health care consumer population presents a number of challenges to patient DSS designers. Patients/consumers vary greatly in age, physical ability, mental ability, computer experience, health care/health condition knowledge, and, as discussed previously, decision-related preferences. They also use different decision strategies when faced with a multiattribute decision tasks such as many of those faced in health care. These decision strategies influence what information the patient (decision maker) wants to see and how he or she views that information. Decision makers frequently use either compensatory or noncompensatory strategies to evaluate alternatives and make a decision. Compensatory strategies are those in which the information about every alternative is weighed and compared (Johnson 1990). Comparatively, in noncompensatory strategies, alternatives may be eliminated after an incomplete search, thereby reducing the cognitive load of the decision task (Johnson 1990). Selection of a decision strategy is related to a number of factors, including age (Johnson 1990) and personality type. In addition to the characteristics of the individual, selection of a decision strategy is also influenced by decision-task characteristics such as the complexity of the decision (Payne 1976) and the type of decision aids available (Todd and Benbasat 2001).

Because of the differing information and decision-support needs of various user populations, designing effective decision support for a broad range of patients/health care consumers is challenging. Therefore, it is imperative that patient DSS designers model the needs of their target user population and design systems to meet those needs. For example, certain patient populations may suffer from limited physical or cognitive abilities, making it vital to create accessible designs that meet their special needs.

30.5 EVALUATING COMPUTERIZED HEALTH CARE APPLICATIONS

Although there has been a proliferation of web-based health care, applications available for consumers and patients, the issue of evaluating applications and guiding users in choosing the best applications has become extremely important. Although evaluation is obviously also important for applications targeted at health care professionals, the issue is not as critical because health care professionals are experts and can exercise their own judgment in the suitability and quality of applications designed to assist them in their work. However, consumers and patients have no or limited basis for exercising such judgment.

Concerned that the growth of the Internet was leading to too much health information with vast chunks of it incomplete, misleading, or inaccurate, Silberg, Lundberg, and Musacchio (1997) proposed the following four standards for websites:

1. Authorship: Authors and contributors, their affiliations, and relevant credentials should be provided.
2. Attribution: References and sources for all content should be listed clearly, and all relevant copyright information noted.
3. Disclosure: Website "ownership" should be prominently and fully disclosed, as should any sponsorship, advertising, underwriting, commercial funding arrangements or support, or potential conflicts of interest. This includes arrangements in which links to other sites are posted because of financial considerations. Similar standards should hold in discussion forums.
4. Currency: dates that content was posted and updated should be indicated.

Murray and Rizzolo (1997) pointed out that

> somehow, just the fact that information is traveling quickly through space and being presented on the computer screen lends it an air of authority which may be beyond its due. Sites with official-sounding names can dupe the inexperienced or uncritical into unquestioned acceptance of the content.

In addition to the standards identified by Silberg, Lundberg, and Musacchio (1997), Murray and Rizzolo (1997), others have cited other criteria for evaluating websites. These include the authority of the author/creator, the accuracy of information and comparability with related sources, the workability (user friendliness, connectivity, search access), the purpose of the resource and the nature of the intended users, whether criteria for information inclusion are stated, the scope and comprehensiveness of the materials, and the uniqueness of the resource.

Jadad and Gagliardi (1998) identified a number of instruments used to provide external ratings of websites. These ratings are used to produce awards or quality ratings, provide seals of approval, identify sites that are featured as the "best of the web" or "best" in a given category, and/or to declare sites as meeting quality standards. They attempted to determine what criteria were used to establish these ratings and to establish the degree of validation of these rating instruments. However, few organizations listed the criteria behind their ratings and none provided information on interobserver validity or construct validity. Jadad and Gagliardi (1998) also discussed the following:

- Whether it is desirable to evaluate information on the Internet due to concerns over freedom of expression, excessive regulatory control, and so on.
- Whether it is possible to evaluate information on the Internet due to the lack of a gold standard for quality information and the controversy surrounding its definition.

The authors also pointed out the differences between Internet information and that of a peer-reviewed journal. These differences include who is exchanging the information (i.e., vendors, healthcare professionals, consumers, insurance companies), the type of format content is delivered in (i.e., Adobe Flash, Javascript, text, sound/video files), and the type of browser and version being used to view the information. They also note that website content is rapidly modified and often linked in a complex web of Internet sites. The complexity of the relationships and links between websites often makes it unclear to the end user that they have left one website for another. They concluded that with respect to external independent evaluation of sites it is not clear:

- Whether evaluation instruments should exist in the first place
- Whether they measure what they claim to measure
- Whether they lead to more good than harm
- Whether users may ever notice, or if they notice, whether they will ignore evidence in support or against desirability, feasibility, or benefits of formal evaluations of health information on the Internet

Robinson et al. (1998) focused more on internal evaluation of applications such as self-evaluation by the sponsors or developers of interactive health communication applications. They pointed out a number of barriers to evaluating these applications. These include the fact that the media and infrastructure underlying applications is in a dynamic state; applications themselves change frequently; many applications are used in situations where a variety of influences on health outcomes exist, few of which are subject to easy assessment or experimental controls; developers lack familiarity with evaluation methods and tools; and developers often believe that evaluation will delay development, increase front-end costs, and have limited impact on sales.

Patrick and Koss (1995) suggested that the effectiveness of consumer health information should be measured by how rapidly and completely desired messages are communicated and how completely intended changes in behavior occur. Saying that Silberg et al.'s criteria (explicit authorship and sponsorship, attribution of sources, and dating of materials) are not enough, Wyatt (1997) provided far more specific direction for evaluating websites. He believes that evaluation of websites should go beyond mere accountability to assess the quality of their content, functions, and likely impact. Evaluating the content should include the following:

- Determining the accuracy of web material by comparing it with the best evidence, that is, for effectiveness of treatment—randomized trials, for risk factors, cohort studies, or for diagnostic accuracy—blinded comparisons of test with a standard.
- Determining timeliness by checking the date on web pages but recognizing that material may not have been current at that time, so need independent comparison with most up-to-date facts is preferable.
- Determining if people can read and understand web material, asking visitors to record satisfaction is unlikely to reveal problems with comprehension since visitors may not realize they misunderstood something or may blame themselves. So, need a minimum reading age for material for the public. However, more accurate to ask users questions based on the web content.

Evaluating the functions of websites should include determining how easy it is to locate a site, how easy it is to locate material within the site, and whether the site is actually used and by whom. For those investing resources in a website, with respect to evaluating the impact, he suggests looking at the impact on clinical processes, patient outcomes, and its cost effectiveness compared with other methods of delivering the same information. He recommended using randomized control trials as the most appropriate method for determining impact. Finally, with respect to evaluation methodology, he pointed out the importance of choosing appropriate subjects (not technology enthusiasts) and the need to make reliable and valid measurements. Wyatt (1997) believed that

> Ideally, investigators would have access to a library of previously validated measurement methods, such as those used for quality of life. However, few methods are available for testing the effect of information resources on doctors and patients, so investigators must usually develop their own and conduct studies to explore their validity and reliability. (p. 1880)

The Science Panel on Interactive Communication and Health (SciPICH) was convened by the Office of Disease Prevention and Health Promotion of the U.S. Department of HHS to examine interactive health-communication technology and its potential impact on the health of the public. The panel was comprised of 14 experts from a variety of disciplines related to interactive technologies and health, including medicine, HCI, public health, communication sciences, educational technology, and health promotion. One of the products of the SciPICH products is an evaluation reporting template (Robinson et al. 1998) for developers and evaluators of interactive health-communication applications to help them report evaluation results to those who are considering purchasing or using their applications. The template has four main sections: (1) description of the application, (2) formative and process evaluation, (3) outcome evaluation, and (4) background of evaluators. The panel defined the three different types of evaluation as follows:

1. Formative evaluation. Used to assess the nature of the problem and the needs of the target audience with a focus on informing and improving program design before implementation. This is conducted prior to or during early application development, and commonly consists of literature reviews, reviews of existing applications, and interviews or focus groups of "experts" or members of the target audience.
2. Process evaluation. Used to monitor the administrative, organizational, or other operational characteristics of an intervention. This helps developers successfully translate the design into a functional application and is performed during application development. This commonly includes testing the application for functionality and may be known as alpha and beta testing.
3. Outcome evaluation. Used to examine an intervention's ability to achieve its intended results under ideal conditions (i.e., efficacy) or under real-world circumstances (i.e., effectiveness), and its ability to produce benefits in relation to its costs (i.e., efficiency or cost effectiveness). This helps developers learn whether the application is successful at achieving its goals and objectives and is performed after the implementation of the application.

Evaluating the effectiveness of web-based applications designed to relay information and/or enable informed decision making is complicated because the "success" of these particular types of applications is (1) not necessarily always related to observable behaviors; (2) based on the quality and usability of the information within the application; and (3) a function of the application itself and the users. In terms of outcome-evaluation applications, Robinson et al. (1998) gave the following examples of the types of questions that such evaluation should address:

1. How much do users like the application?
2. How helpful/useful do users find the application?
3. Do users increase their knowledge?
4. Do users change their beliefs or attitudes (i.e., self-efficacy, perceived importance, intentions to change behavior, and satisfaction)?
5. Do users change their behaviors (i.e., risk factor behaviors, interpersonal interactions, compliance, and use of resources)?
6. Are there changes in morbidity or mortality (i.e., symptoms, missed days of school/work, physiologic indicators)?
7. Are there effects on cost/resource utilization (i.e., cost-effectiveness analysis)?
8. Do organizations or systems change (i.e., resource utilization, and effects on "culture")?

However, for websites designed primarily to provide information or enable informed decision making, only the first three questions apply. Other potential outcomes related to change in behavior might be applicable depending on the nature of decisions made. Consequently, the evaluation of these types of web guides needs to focus on the use of the guides (assuming that users who "like" an application will use it more than those who do not), the usefulness of the guides, the usability of the guides, the ability of the guide to increase knowledge, and the contribution of the guide to decision making. These elements can be integrated into a conceptual framework as shown in Figure 30.1. Part of this framework draws from the Agency for Health Care Policy and Research reported on CHI and patient decision making (1997) as well as Sainfort and Booske (1996).

The framework suggests that to evaluate the effectiveness of web-based health applications, at least three perspectives (at the bottom of the figure) can be taken individually or in combination: (1) the consumer, (2) the site sponsor, and (3) outside experts. The framework posits that the characteristics of the system under evaluation primarily influence accessibility and usability of information. Then, accessibility, in conjunction with consumer/patient characteristics, influence actual access to information. In turn, the usability of this information, again in conjunction with consumer characteristics, will influence actual use of information by the consumer. Use of information is a complex construct. The framework emphasizes three main uses of information: (1) knowledge (inquiry, verify, learn, augment, etc.), (2) decision making,

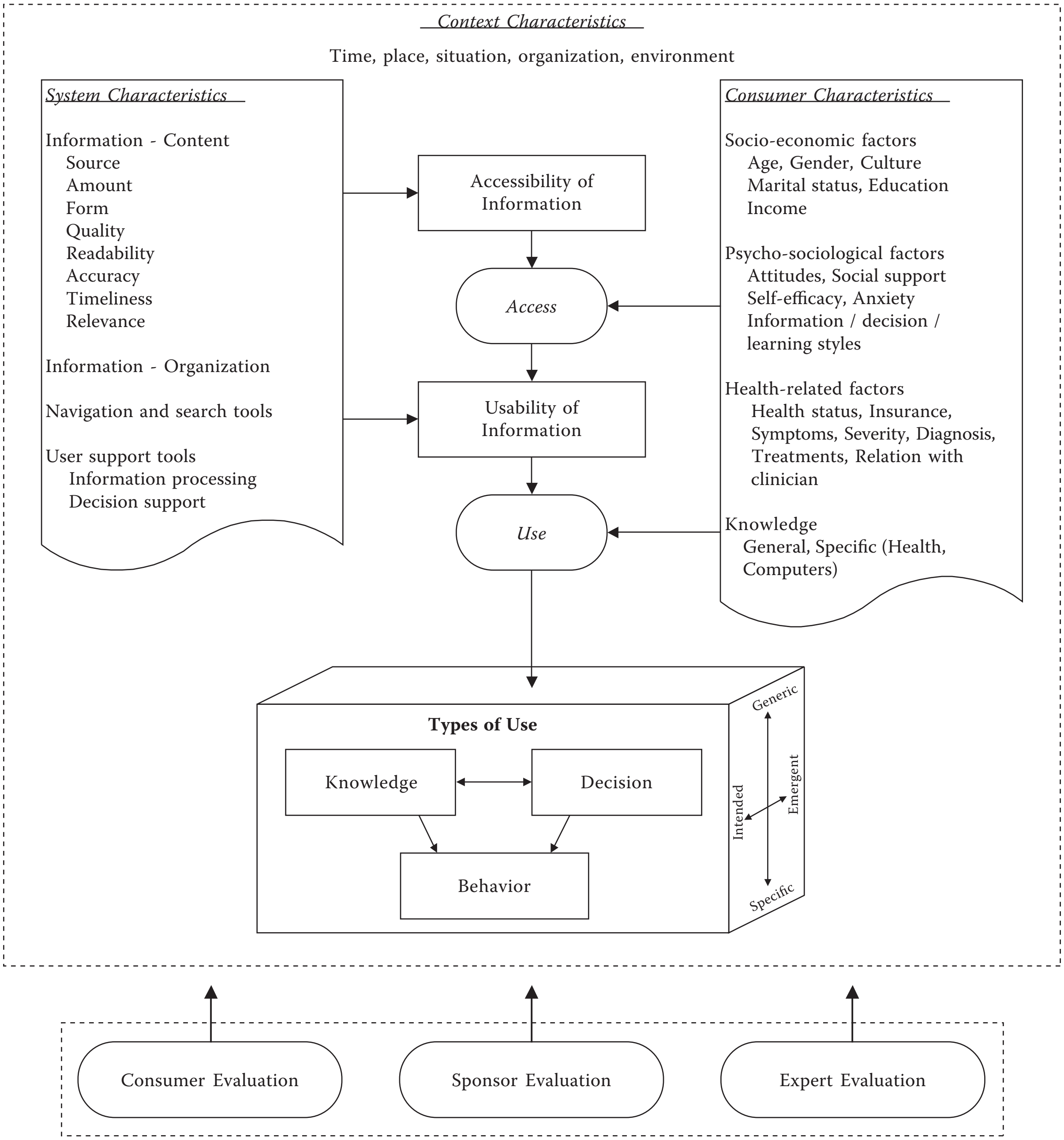


FIGURE 30.1 Conceptual framework.

and (3) behavior (intentions and actual behavior change). All three uses are generally interconnected, with behavior usually following knowledge and/or decision making (whether explicitly or implicitly). Also emphasized in the framework is the fact that "use" can be generic (common to most web-based health applications) or specific (to the application) and also that use of information accessed can be intended (i.e., the information was sought to accomplish a specific purpose or use) or emergent (i.e., a piece of information accessed triggers a new use). Finally, the framework shows, surrounding this entire process, that the context (the situation, time, place, organization, etc.) can influence all key elements: accessibility, access, usability, and use. Consumer evaluations are formed because of the experiences they have using the system over time. Expert evaluation usually involves assessment of the system itself and its content, as well as an assessment of its (anticipated) impact on users. Sponsors evaluation involves both consumer and expert evaluations as well as consideration of organizational resources expended in the design, development, implementation, and operation of the system.

It is important to differentiate information access and quantity from use and usefulness (Booske and Sainfort 1998). Indeed, while many studies primarily address attempts to measure the "quantity" of information used, others have acknowledged the need to consider the "quality" of information. The introduction of the web as a source of information has introduced a number of additional considerations in evaluation of the use and usefulness of information. The most common measure of information use on the web is the number of "hits" or "visits" to a particular page. Most of these counts do not differentiate between multiple visits by a single user versus multiple users nor do they consider the amount of time spent on a particular page, that is, whether the page is merely used as a link to elsewhere or as a source of information itself.

In describing three general approaches to information systems evaluation (i.e., the system's output, behavior, and architecture), Orman (1983) defined the quality and quantity of information as the relevant variables in defining an information system's output. However, he went on to say that both quantity and quality of information were "of considerable theoretical interest but of little practical value since neither can be defined or measured with acceptable precision" (p. 309). Orman defined the quality of information produced by a system in terms of "its contributions to the quality of the decisions it aids" and points out that this is "highly influenced by the style and the behavior of the information user and the state of the environment" (p. 309).

This ties with the concept of the "value" of information, an important part of traditional decision and economic analysis. Within the field of decision science, the primary use of information is to help make decisions. However, an examination of literature from other disciplines has shown that there are other ways to conceptualize the value, and thus the effectiveness of information and its usefulness (Orman 1983). Although it can be thought that the term "information usefulness" could be an analogue for value of information, within the field of HCI, the concept of usefulness includes additional dimensions. The concepts of "perceived usefulness" and "usability" of information systems are both receiving more attention as more systems are developed for novice users. For example, Davis, Bagozzi, and Warshaw (1999) developed scales to measure perceived usefulness and ease of use to evaluate specific software in the work setting. They defined perceived usefulness as "a prospective user's subjective probability that using a specific application system will increase his or her job performance within an organizational context." Ease of use was defined as "the degree to which the prospective user expects the target system to be free of effort." Using a seven-point Likert scale, potential users of software responded to six items addressing perceived usefulness and six to assess perceived ease of use.

Of particular relevance in considering the usability, and potentially the usefulness, of information is the manner of display. Although technological advances continue to increase the number of possible methods for disseminating information, the primary output media for health care information are still print, video, and sound. Print information is often categorized as text, tables, graphs, and figures. There is a quite extensive body of literature comparing the effectiveness of displaying information in tables versus graphs (Jarvenpaa and Dickson 1988; Dahlberg 1991 provided comprehensive summaries of this empirical work). Many of these studies rely on elicited "directed interpretations" where subjects are given specific questions about the information in tables and graphs. In contrast, a more recent study by Carswell and Ramzy (1997) elicited spontaneous interpretation of a series of tables, bar graphs, and line graphs to find out "what information subjects choose to take away from a display rather than their ability to extract information when promoted."

Within the context of decision making, Schkade and Kleinmuntz (1994) found that different characteristics of information display affected aspects of choice processes. They found that organization of information (such as a matrix vs. a list) influenced organization acquisition; form (numeric vs. linguistic) influenced information combination and evaluation; while sequence had only a limited effect on acquisition. Johnson, Payne, and Bettman (1988) and others have provided evidence of preference reversals because of different information displays.

Many of the issues surrounding the use of the Internet are issues of usability. Usability is a central notion of the HCI field. Simply put, Shackel (1991) defined usability as the "the capability to be used by humans easily and effectively," whereas his more formal operational and goal-oriented definition said, "for a system to be usable, the following must be achieved":

Effectiveness. The required range of tasks must be accomplished better than some required level of performance (i.e., in terms of speed and errors); by some required percentage of the specified target range of users; and within some required proportion of the range of usage environments.

Learnability. Within some specified time from commissioning and start of user training; based on some specified amount of user training and support; and within some specified relearning time each time for intermittent users.

Flexibility. With flexibility allowing adaptation to some specified percentage variation in tasks and/or environments beyond those first specified.

Attitude. And within acceptable levels of human cost in terms of tiredness, discomfort, frustration, and personal effort; so that satisfaction causes continued and enhanced usage of the system. (p. 48)

In an attempt to place system usability in relation to other system concepts, Shackel (1991) suggested that

> utility (will it do what is needed functionally?), usability (will the users actually work it successfully?) and likeability (will the users feel it is suitable?) must be balanced in a trade-off against cost (what are the capital and running costs and what are the social and organizational consequences?) to arrive at a decision about acceptability (on balance the best possible alternative for purchase). (p. 50)

These concepts are all relevant in a decision to purchase or accept an information system, but may not be all relevant when the resource or product under consideration is information. As Orman (1983) pointed out the value of an information system is different from the value of its information content, "just as the value of a candy machine is different from the value of the candy it dispenses." Furthermore, usability of information is a necessary but not sufficient criterion for information to be useful. (p. 312).

In the context of informing health care consumer decisions, Hibbard, Slovic-Jewett, and Jewett (1997) asked how much information is too much. In response to this question, they determined that the critical element is the ability to interpret and integrate information items: integration is a very difficult cognitive process. They suggest that for a consumer, more information is not necessarily better and that the simple provision is not sufficient when the information is complex. In complex-decision situations, it is important to pay particular attention to human information processing capabilities and differences across individuals. For example, specifically addressing information presentation format, Togo and Hood (1992) found a significant interaction effect between gender and format. Personality differences have also been found to contribute to variation in information processing. The way people gather and evaluate information is at least in part based on their psychological type (Slocum and Hellriegel 1983). Some people are driven to know details before decisions are made, whereas others feel more comfortable assuming what is not known. Individuals vary on how they receive messages, seek information, organize information, and process information. One approach to categorizing cognitive style, the Myers-Briggs Type Indicator, is based on Jung's typology (Myers 1987).

Other models of learning styles can be used to differentiate among individuals and their preferred methods or strategies for taking in and processing information. Felder (1996) provided a useful summary of these models. For example, Kolb's learning-style model classified people as having a preference for (1) concrete experience or abstract conceptualization (how they take information in) and (2) active experimentation or reflective observation (how they internalize information). The Hermann Brain Dominance Instrument classified people in terms of their relative preferences for thinking that are based on the task-specialized functioning of the physical brain, that is, left brain versus right brain, cerebral versus limbic. The Felder-Silverman learning style model classified learners on five dimensions: (1) sensing or intuitive, (2) visual or verbal, (3) inductive or deductive, (4) active or reflective, and (5) sequential or global.

In addition to individual differences, other factors that can affect information processing reflect characteristics of the information itself (John and Cole 1986), that is, information quantity, information source, information format (mode of presentation, organization, order), information complexity, nature of access (i.e., voluntary vs. mandatory), instructions in the use of the information, response formats (i.e., recognition, recall, judgment, choice), and the interface itself.

Understanding cognitive processes involved in accessing, processing, interpreting, and using health care information is critical to the successful design and implementation of web-based health care applications. This particular point extends to applications targeted at providers and other health care professionals. In looking at the impact of computer-based patient record systems on data collection, knowledge organization, and reasoning, Patel et al. (2000) indicated that exposure to computer-based patient records was associated with changes in physicians' information gathering and reasoning strategies. They concluded that such technology could have profound influence in shaping cognitive behavior. In such systems, the HCI itself can have a strong influence on information gathering, processing, and reasoning strategies. Recently, Patel, Arocha, and Kaufman (2001) surveyed literature on aspects of medical cognition and suggested that

> cognitive sciences can provide important insights into the nature of the processes involved in human–computer interaction and help improve the design of medical information systems by providing insight into the roles that knowledge, memory, and strategies play in a variety of cognitive activities. (p. 324)

30.6 FUTURE OPPORTUNITIES AND CHALLENGES

Information and communication technologies have only begun to impact the health care industry. Regarding the use of the Internet, a number of applications will be at the leading edge. Mittman and Cain's (1999) prediction that these applications would include consumer health-information services, online support groups for patients and caregivers, health care-provider-information services, provider-patient e-mail, communications infrastructure and transaction services, and EMRs appears to be true. Similarly, the barriers they predicted would impede or slow down the development of the Internet in health care have largely proven true as well: security concerns, weaknesses inherent to web interfaces (especially browsers, search-engine technology, and inability to interact with legacy systems), mixed or lack of quality of information, physician ambivalence, disarray of current health care-information systems, lack of resources for web development, and lack of unified standards for electronic communications and transactions.

For information technology to fully impact health care delivery with a focus on both patients and providers, these barriers need to be overcome. We believe that the greatest benefits will be reached in the short term by focusing on the following five areas.

30.6.1 Supporting/Enhancing the Patient-Provider Interaction

The Internet will become a critical element of the physician-patient encounter. Organizational websites will evolve from publishing generic consumer content to providing personalized, online services to all consumers (individuals, patients, providers, etc.). The Internet needs to provide for technology that truly supports two-way interaction between patients and providers and directly impacts care delivery.

Effective technologies need to be developed to fully support, enhance, and extend the patient–physician interaction so as to increase the efficiency of care delivery, increase the quality of care received by patients, and increase the effectiveness of the work performed by the physician. This latter point is important since a study by Linzer et al. (2000) showed that time stress, defined as reports by the physician that they needed more time for patients than they were allotted, was significantly related to burnout, job dissatisfaction, and patient-care issues. This study was performed with a national sample (n=5, 5704) of physicians in primary-care specialties and medical and pediatric subspecialties. It demonstrates that the ability to spend sufficient and quality time with each patient is critical to ensure long-term job satisfaction, avoid physician burnout, and increase quality of patient care.

In developing such technologies, as we mentioned earlier, one would need a full understanding of the various cognitive processes involved in information gathering, knowledge acquisition and organization, reasoning strategies, and decision making. Of particular importance is the recognition of various users with various characteristics potentially using technologies in a variety of situations, contexts, organizations, and environments.

30.6.2 Supporting/Enhancing Collaborative Work among Providers

The nature of the physician/associate provider (nurse, physician assistant) relationship is evolving from mere interaction to true collaboration with technologies allowing associate providers to increase involvement in clinical decision making and implementation. Technologies to support collaborative work environment in fast-pace, mobile environment are needed.

As mentioned earlier, the health care work environment is very complex and involves a variety of health care professionals, all with varying needs for information, knowledge, and support. Designing technologies supporting both individual needs and collaboration among individuals presents a number of challenges. A study by Jacko, Sears, and Sorensen (2001) showed that different health care professionals (physicians, pharmacists, nurses) exhibit very different patterns of use of the Internet for clinical purposes and have very different perceptions of needed enhancements to support their respective needs. The patterns differed in terms of the range and type of information as well as the depth and specificity of information.

Health care organizations will be faced with technology integration challenges and will make use of the Internet and Intranet technologies to manage and organize the variety of applications in use by providers and their patients. Of critical importance for health care organizations will be the design of systems to support clinical decisions, knowledge management, organizational learning, administrative transactions, and supply chain.

30.6.3 Developing and Utilizing New Information and Communication Technologies

Wireless, handheld, and nanotechnologies will drive higher adoption rates of the Internet in clinical settings, patient monitoring, and disease management. Both the number of interactive wireless device users and the number of devices will increase significantly in the next few years. New applications for such technologies need to be developed, based on fundamental research conducted to investigate usage of such devices.

Mobile computing and wireless technologies will increasingly become an important part of health care's information technology. Turisco and Case (2001) initially reported that mobile computing applications for health care began with reference tools and then moved to transaction-based systems to automate simple clinical and business tasks. In 2004, Goldsmith argued that mobile computing was a promising emerging IT innovation with the potential to fundamentally transform the way health care services are delivered. We have in fact seen a transformation in health care due to mobile computing with improvements in CPOE (Junglas, Abraham, and Ives 2009) and patient monitoring (Sneha and Varshney 2009). Mobile computing technologies will continue to advance due to increasing improvements in bandwidth that allow medical images to be transmitted easily, and a variety of new vendor applications and wireless initiatives.

30.6.4 Designing and Utilizing Adaptive Human–Computer Interactions

The development and integration of information and communication technologies designed to support, facilitate, and enhance patient/provider interactions are critical. In particular, research is needed to design optimal HCIs. A dual focus on (1) increasing job satisfaction and effectiveness for the medical personnel and (2) increasing quality of care and safety for patients is needed. To support these objectives, the HCI must possess at least the following four qualities:

1. Multimodal, that is, have the ability to display and accept information in a combination of visual, aural, and haptic modes
2. Personalized, that is, tailored to respond in a manner best suited to the current user and his or her needs
3. Multisensor, that is, have the ability to detect and transmit changes in the user and/or situation
4. Adaptive, that is, have the ability to change its behavior in real time to accommodate user preferences, user disabilities, and changes in the situation/environment

Optimal interfaces are critical to virtually all applications connecting people to information technologies and people to people via information and communication technologies. Current research is underway to develop intelligent adaptive multimodal interface systems. In the future, interfaces will automatically adapt themselves to the user (capabilities, disabilities, etc.), task, dialogue, environment, and input/output modes to maximize the effectiveness of the HCI.

This is especially critical in health care for both types of users: consumers/patients and providers. Consumers/patients present a number of challenges regarding the interface. In addition to presenting different personal and socio-demographic characteristics, users in health care present varying degrees of health status: healthy consumers, newly diagnosed patients, chronically ill patients, and/or their caregivers will use the same devices in potentially very different ways. Similarly, different providers will have very different characteristics and will perform a multitude of varying and highly dynamic tasks in different contexts and environments.

30.6.5 Moving to e-Health

The opportunities for improving service and decreasing cost via e-commerce technologies and the supply chain are tremendous. In addition, modern information and communication technologies (including sensors, wireless communication, and implant technologies) will enable electronic delivery of health care. This is far more comprehensive than the mere electronic delivery of health information to patients and providers and includes new developments such as telemedicine and virtual reality. Krapichler et al. (1999) claimed that “virtual environments are likely to be used in the daily clinical routine in [the] medicine of tomorrow” (p. 448). However, a number of barriers will need to be overcome. Organizational barriers to e-Health include infrastructure, organization, culture, and strategy, systems integration, and workflow integration. Technological barriers include integration and security, interface design, connectivity, speed, reliability, and usability issues.

The evolution toward e-Health will involve a number of major changes. For example, organizational websites will evolve from publishing generic consumer content to providing personalized, online interactive services to profitable patient segments. Delivery organizations will focus on direct-to-patient relationship building, migrating to a health system truly centered on patients and communities. Communities of interest will rapidly expand to become a force in health care navigation. Wireless and handheld technologies will drive higher adoption rates of the Internet in clinical settings. The Internet will become a critical element of the physician–patient encounter and will support and enhance the patient–provider interaction. Social media will play an increasing role, as well, but further exploration into the appropriate role of social media in health care education and care delivery is needed. These changes will lead to a restructuring of the health care industry. Digital health plans will emerge and compete or threaten the traditional health care-payer business model. Virtual networks will begin to emerge around specialty services to provide efficient personalized health care.

ACKNOWLEDGMENTS

The authors recognize the contributions of Dr. Bridget Booske, who co-authored the first and second edition versions of this chapter.

REFERENCES

Acosta, D., V. Patkar, M. Keshtgar, and J. Fox. 2010. Challenges in delivering decision support systems: The MATE experience. In *Knowledge Representation for Healthcare. Data, Processes and Guidelines*, Vol. 5943, ed. D. Riano, A. T. Teije, S. Miksch, and M. Peleg, 124–40. Berlin/Heidelberg: Springer. Print. Lecture Notes in Computer Science.

Adams, W. G., A. M. Mann, and H. Bauchner. 2003. Use of an electronic medical record improves the quality of urban pediatric primary care. *Pediatrics* 111:626–32.

Adnan, M., J. Warren, and M. Orr. 2009. Enhancing patient readability of discharge summaries with automatically generated hyperlinks. *Health Care Inf Rev Online* 13:21–7. http://www. hinz.org.nz/uploads/file/Journal_Dec09/Adnan_P21.pdf (accessed March 19, 2011).

Adnan, M., J. Warren, M. Orr, A. Ewens, J. Scott, and S. Trubshaw. 2010. The quality of electronic discharge summaries for post-discharge care: Hospital panel assessment and IT to support improvement. *Health Care and Inf Rev Online* 14:8–17. http://www.hinz.org.nz/uploads/file/Journal_Dec10/Adnan_P8.pdf (accessed March 19, 2011).

American Recovery and Reinvestment Act. 2009. Recovery.gov. frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=111_cong_bills&docid=f:h1enr.pdf (accessed March 2, 2011).

Anderson, J., and E. Balas. 2006. Computerization of primary care in the United States. *Int J Health Inform Syst Inform* 1:1–23.

Anvari, M., C. McKinley, and H. Stein. 2005. Establishment of the world’s first telerobotic remote surgical service for provision of advanced laparoscopic surgery in a rural community. *Ann Surg* 241:460–4.

Bates, D. W., D. L. Boyle, and J. M. Teich. 1994. Impact of computerized physician order entry on physician time. In *Paper presented at the Annual Symp Comput Appl Med* Care. Bethesda, MD: AMIA.

Bates, D. W., M. Cohen, L. L. Leape, M. Overhage, M. M. Shabot, and T. Sheridan. 2001. Reducing the frequency of errors in medicine using information technology. *JAMIA* 8(4):299–308.

Bates, D. W., J. M. Teich, J. Lee, D. Seger, G. J. Kuperman, N. I. Ma'Luf, D. Boyle, and L. Leape. 1999. The impact of computerized physician order entry on medication error prevention. *J Am Med Inform Assoc* 6(4):313–21.

Bekker, H. L., J. Hewison, and J. G. Thornton. 2003. Understanding why decision aids work: Linking process with outcome. *Patient Educ Couns* 50:323–9.

Benton Foundation. 1999. *Networking for Better Care: Health Care in the Information Age*. http://www.benton.org/Library/health/ (accessed December 5, 2001).

BlueCross BlueShield Association. 2008. Medical cost reference guide. http://www.bcbs.com/blueresources/mcrg/ (accessed February 22, 2011).

Blumenthal, D. 2009. *The Federal Role in Promoting Health Information Technology*. New York: The Commonwealth Fund Perspectives on Health Reform. http://www.commonwealthfund.org/Content/Publications/Perspectives-on-Health-Reform-Briefs/2009/Jan/The-Federal-Role-in-Promoting-Health-Information-Technology.aspx (accessed March 2, 2011).

Boonstra, A., and M. Broekhuis. 2010. Barriers to the acceptance of electronic medical records by physicians from systematic review to taxonomy and interventions. *BMC Health Serv Res* 10(1):231. http://www .biomedcentral.com/1472-6963/10/231 (accessed March 1, 2011).

Booske, B. C., and F. Sainfort. 1998. Relationships between quantitative and qualitative measures of information use. *Int J Hum Comput Interact* 10(1):1–21.

Boulos, K., I. Maramba, and S. Wheeler. 2006. Wikis, blogs and podcasts: a new generation of web-based tools for virtual collaborative clinical practice and education. *BMC Medical Education* 6(1):1–8.

Branstetter, B. F. 2007. Basics of imaging informatics: Part 11. *Radiology* 243(3):656–67. doi:10.1148/radiol.2433060243.

Carswell, C. M., and C. Ramzy. 1997. Graphing small data sets: Should we bother? *Behav Inf Technol* 16(2):61–70.

Chassin, M. R., and R. W. Galvin, The National Roundtable on Health Care Quality. 1998. The urgent need to improve health care quality. *JAMIA* 280(11):1000–5.

Chaudhry, S. I., J. A. Mattera, J. P. Curtis, J. A. Spertus, J. Herrin, Z. Lin, and H. M. Krumholz. 2010. Telemonitoring in patients with heart failure. *N Engl J Med* 363(24):2301–09. doi:10.1056/NEJMoa1010029

Cheng, C. H., M. K. Goldstein, E. Geller, and R. E. Levitt. 2003. The Effects of CPOE on ICU workflow: An observational study. In *AMIA Annual Symposium*, 150–44. Washington, DC: AMIA.

Cohen, S. B., K. D. Grote, W. E. Pietraszek, and F. Laflamme. 2010. Increasing consumerism in healthcare through intelligent information technology. *Am J Manag Care* 16, SP37–P43. http://www .ajmc.com/media/pdf/AJMC_10decHIT_CohenSP37to43.pdf (accessed March 14, 2011).

Congressional Budget Office. 2007. *The Long-Term Outlook for Health Care Spending*. http://www.cbo.gov/ftpdocs/87xx/doc8758/MainText.3.1.shtml (accessed February 22, 2011).

Cordero, L., L. Kuehn, R. R. Kumar, and H. S. Mekhjian. 2004. Impact of computerized physician order entry on clinical practice in a newborn intensive care unit. *J Perinatology* 24(2):88–93.

Dahlberg, T. 1991. Effectiveness of report format and aggregation: An approach to matching task characteristics and the nature of formats. In *Acta Academie Oeconomicae Helsingiensis Series A:76*. Helsinki, Finland: Helsinki School of Economics and Business Administration.

Davis, F. D., R. P. Bagozzi, and P. R. Warshaw. 1999. User acceptance of computer technology: A comparison of two theoretical models. *Manage Sci* 35:982–1003.

Edwards, P. J., K. P. Moloney, J. A. Jacko, and F. Sainfort. 2008. Evaluating usability of a commercial electronic health record: A case study. *Int J Hum Comput Stud* 66(10):701–58.

Ereso, A. Q., P. Garcia, E. Tseng, G. Gauger, H. Kim, M. M. Dua, G. P. Victorino, and T. S. Guy. 2010. Live transference of surgical subspecialty skills using telerobotic proctoring to remote general surgeons. *J Am Coll Surg* 211(3):400–11. http://www.journalacs.org/article/S1072-7515%2810%2900374-1/abstract (accessed March 14, 2011).

Eysenbach, L. et al. eds. 2005. Design and evaluation of consumer health information web sites. *Consumer Health Informatics* 34–60.

Eysenbach, G. 2005. Design and evaluation of consumer health information web sites consumer health informatics. In Lewis, D., Eysenbach, G., Kukafka, R., Stavri, P. Z., Jimison, H. B., Hannah, K. J., Ball, M. J., Hannah, K. J., and Ball, M. J., eds., *Consumer Health Informatics, Health Informatics*, 34–60. New York: Springer.

Felder, R. M. 1996. Matters of Style. *ASEE Prism* 6(4):18–23.

Ferguson T. 1997. Health online and the empowered medical consumer. *Jt Comm J Qual Improv* 23(5):251–7. PubMed PMID: 9179717.

Fox, S. 2006. Online Health Search 2006 | Pew Research Center's Internet & American Life Project. Pew Research Center's Internet & American Life Project. http://www.pewinternet.org/Reports/2006/Online-Health-Search-2006.aspx (accessed March 14, 2011).

Gardner, R. M., and M. Shabot. 2001. Patient monitoring systems. In *Medical informatics: Computer applications in health care and biomedicine*, 2nd ed., ed. E. H. Shortliffe, L. E. Perreault, G. Wiederhold, and L. Fagan, 443–84. New York: Springer.

Gibbons, M. C., R. F. Wilson, L. Samal, C. U. Lehmann, K. Dickersin, H. P. Lehmann, H. Aboumatar et al. 2009. *Impact of Consumer Health Informatics Applications*. Evidence Report/Technology Assessment No. 188. (Prepared by Johns Hopkins University Evidence-based Practice Center under contract No. HHSA 290-2007-10061-I). AHRQ Publication No. 09(10)-E019. Rockville, MD: Agency for Healthcare Research and Quality.

Goldsmith, J. 2004. Technology and the boundaries of the hospital: Three emerging technologies. *Health Affairs* 23(6):149–56. doi:10.1377/hlthaff.23.6.149.

Greenes, R. A., and J. F. Brinkley. 2001. Imaging systems. In *Medical informatics: Computer applications in health care and biomedicine*, 2nd ed., ed. E. H. Shortliffe, L. E. Perreault, G. Wiederhold, and L. Fagan, 485–538. New York: Springer.

Gustafson, D. H., R. Hawkins, E. Boberg, S. Pingree, R. E. Serlin, F. Graziano, and C. L. Chan. 1999. Impact of a patient-centered, computer-based health information/support system. *Am J Prev Med* 16(1):1–9.

Gustafson, D. H., R. Hawkins, S. Pingree, F. McTavish, N. K. Arora, J. Mendenhall, D. F. Cella et al. 2001. Effect of computer support on younger women with breast cancer. *J Gen Intern Med.* 16(7):435–45.

Han, Y. Y., J. A. Carcillo, S. T. Venkataraman, R. S. B. Clark, R. S. Watson, T. C. Nguyen, H. Bayir, and R. A. Orr. 2005.

Unexpected increased mortality after implementation of a commercially sold computerized physician order entry system. *Pediatrics* 116(6):1506–12.

Harris, J. 1995. Consumer health information demand and delivery: A preliminary assessment. In *Partnerships for Networked Health Information for the Public*. Rango Mirage, California. Summary Conference Report. Office of Disease Prevention and Health Promotion, U.S. Department of Health and Human Services, Washington, DC 20201.

Health Edge - Watch. 2011. Cleveland Clinic. http://my.clevelandclinic.org/h.ealth_edge/vodcasts.aspx (accessed March 14, 2011).

Hibbard, J. H., P. Slovic, and J. J. Jewett. 1997. Informing consumer decisions in health care: Implications from decision-making research. *Milbank Q* 75(3):395–414.

Hoffman, C., D. P. Rice, and H. Y. Sung. 1996. Persons with chronic conditions: Their prevalence and costs. *JAMA* 276(18):1473–79.

Holden, R. J. 2010. Physicians' beliefs about using EMR and CPOE: In pursuit of a contextualized understanding of health IT use behavior. *Int J Med Inf* 79(2):71–80. doi: 10.1016/j.ijmedinf.2009.12.003.

Huang, H. K. 2010. PACS and Imaging Informatics Basic Principles and Applications, 2nd ed. Hoboken, NJ: Wiley-Blackwell.

Hughes, B., I. Joshi, H. Lemonde, and J. Wareham. 2009. Junior physician's use of web 2.0 for information seeking and medical education: A qualitative study. *Int J Med Inf* 78(10):645–55.

Inglis, S. C., R. A. Clark, F. A. McAlister, J. Ball, C. Lewinter, D. Cullington, S. Stewart, and J. G. Cleland. 2010. Structured telephone support or telemonitoring programmes for patients with chronic heart failure. *Cochrane Database Syst Rev* 8:CD007228. Review. PubMed PMID: 20687083.

Institute of Medicine. 2000. *To err is Human: Building a Safer Health System*. Washington, DC: National Academy Press.

Institute of Medicine. 2001. *Crossing the quality chasm: A new health system for the 21st century*. Washington, DC: National Academy Press.

Jacko, J. A. 2011. Narrow the gap in health literacy. *Nature* 470:328.

Jacko, J. A., A. Sears, and S. J. Sorensen. 2001. A framework for usability: Healthcare professionals and the Internet. *Ergonomics* 44(11):989–1007.

Jadad, A. R., and A. Gagliardi. 1998. Rating health information on the Internet: Navigating to knowledge or to Babel? *JAMA* 279(8):611–14.

Jarvenpaa, S. L., and G. W. Dickson. 1988. Graphics and managerial decision making: Research based guidelines. *Commun ACM* 31(6):764–74.

John, D. R., and C. A. Cole. 1986. Age differences in information processing: Understanding deficits in young and elder consumers. *J Consum Res* 13(12):297–315.

Johnson, M. M. S. 1990. Age differences in decision making: A process methodology for examining strategic information processing. *J Gerontology* 45(2):75–8.

Johnson, E. J., J. W. Payne, and J. R. Bettman. 1988. Information displays and preference reversals. *Organ Behav Hum Decis Processes* 42:1–21.

Junglas, I., C. Abraham, and B. Ives. 2009. Mobile technology at the frontlines of patient care: Understanding fit and human drives in utilization decisions and performance. *Decis Support Syst* 46(3), 634–47. doi: 10.1016/j.dss.2008.11.012.

Kadmon, G., E. Bron-Harlev, E. Nahum, O. Schiller, G. Haski, and T. Shonfeld. 2009. Computerized order entry with limited decision support to prevent prescription errors in a PICU. *Pediatrics* 124(3):935–40. doi:10.1542/peds.2008-2737.

Kaelber, D. C., S. Shah, A. Vincent, E. Pan, J. Hook, D. Johnston, D. Bates, and B. Middleton. 2008. *The Value of Personal Health Records. Center for Information Technology Leadership*. Charleston: Mass.

Karsh, B., J. W. Beasley, M. E. Hagenauer, and F. Sainfort. 2001. Do electronic medical records improve the quality of medical records? In *Systems, Social and Internationalization Design Aspects of Human-Computer-Interaction*, ed. M. J. Smith and G. Salvendy, 908–12. Mahwah, NJ: Lawrence Erlbaum Associates.

Keselman A., T. Tse, J. Crowell, A. Browne, L. Ngo, and Q. Zeng. 2007. Assessing consumer health vocabulary familiarity: An exploratory study. *J Med Internet Res* 9(1):e5. http://www.jmir.org/2007/1/e5/ (accessed March 19, 2011).

Kvedar, J. 2008. Community Healthcare Discussion Report. Center for Connected Health. www.connected-health.org/media/213517/obama%20transition%20team%20recommendations.pdf. Retrieved 4 November 2011.

Koppel, R., J. P. Metlay, A. Cohen, B. Abaluck, A. R. Localio, S. E. Kimmel, and B. L. Strom. 2005. Role of computerized physician order entry systems in facilitating medication errors. *JAMA* 293(10):1197–203.

Krapichler, C., M. Haubner, A. Losch, D. Schuhmann, M. Seemann, and K. H. Englmeier. 1999. Physicians in virtual environments—Multimodal human-computer interaction. *Interact Comput* 11:427–52.

Krohn, R. 2008. Health information exchanges—what's working? *J Healthc Inf Manag* 22(3):7–8. PubMed PMID:19267023.

Kumar, S., and C. Chandra. 2001. A healthy change. *IIE Solutions* 33(3):29–33.

Kuperman, G. J., A. Bobb, T. H. Payne, A. J. Avery, T. K. Gandhi, G. Burns, D. C. Classen, and D. W. Bates. 2007. Medication-related clinical decision support in computerized provider order entry systems: A review. *J Am Med Inf Assoc* 14:29–40.

Legler, J. D., and R. Oates. 1993. Patients' reactions to physician use of a computerized medical record system during clinical encounters. *J Family Pract* 37(3):241–44.

Lindberg, D. A. B., and B. L. Humphreys. 1998. Medicine and health on the Internet: The good, the bad, and the ugly. *JAMA* 280(15):1303–04.

Linzer, M., T. R. Konrad, J. Douglas, J. E. McMurray, D. E. Pathman, E. S. Williams, M. D. Schwartz et al. 2000. Managed care, time pressure, and physician job satisfaction: Results from the physician worklife study. *J Gen Int Med* 15(7):441–50.

Llewellyn-Thomas, H. A., M. J. McGreal, and E. C. Thiel. 1995. Cancer patients' decision making and trial-entry preferences: The effects of "framing" information about short-term toxicity and long-term survival. *Med Decis Making* 15(1):4–12.

Lyman, J. A., W. F. Cohn, M. Bloomrosen, and D. E. Detmer. 2010. Clinical decision support: Progress and opportunities. *JAMIA* 17(5):487–92.

Mark, T. L., G. Johnson, B. Fortner, and K. Ryan. 2008. The benefits and challenges of using computer-assisted symptom assessments in oncology clinics: results of a qualitative assessment. *Technol Cancer Res Treat* 7(5):401–6.

Masterson, L. 2008. Two phrases from AHIP: Consumerism and interactivity. HealthLeaders Media for Healthcare Executives—HealthLeaders Media. http://healthplans.hcpro.com/print/HEP-214109/Two-Phrases-from-AHIP-Consumerism-and-Interactivity (accessed February 18, 2011).

Mekhjian, H. S., R. R. Kumar, L. Kuehn, T. D. Bentley, P. Teater, A. Thomas et al. 2002. Immediate benefits realized following implementation of physician order entry at an academic medical center. *JAMIA* 9(5):529–39.

Mittman, R., and M. Cain. 1999. *The Future of the Internet in Health Care*. Oakland, CA: California Healthcare Foundation.

Molenaar, S., M. A. G. Sprangers, F. C. E. Postma-Schuit, E. J. T. Rutgers, J. Noorlander, J. Hedriks, and H. De Haes. 2002. Feasibility and effects of decision aids. *Med Decis Making* 20(1):112–27.

Murray, C. J. L., D. Phil, and J. Frenk. 2010. Ranking 37th—measuring the performance of the U.S. health care system. Health policy and reform. *N Engl J Med*. http://healthpolicyandreform.nejm.org/?p=2610 (accessed February 22, 2011).

Murray, P. J., and M. A. Rizzolo. 1997. Reviewing and evaluating websites—some suggested guidelines. *Nursing Stand Online* 11(45). http://www.nursing-standard.co.uk/vol11-45/ol-art.htm (accessed July 30, 1997).

Musen, M. A., Y. Shahar, and E. H. Shortliffe. 2006. Clinical decision-support systems. In *Biomedical Informatics*, 2nd ed., ed. E. H. Shortliffe and J. J. Cimino, 698–736. New York: Springer. http://dx.doi.org/10.1007/0-387-36278-9_20. (accessed February 22, 2011).

Myers, I. B. 1987. *Introduction to Type*. Palo Alto, CA: Consulting Psychologists Press.

National Alliance for Health Information Technology. 2008. Office of the National Coordinator for Health Information Technology. Defining Key Health Information Technology Terms. United States Department of Health and Human Services. www.hhs.gov/healthit/documents/m20080603/10_2_hit_terms.pdf (accessed March 19, 2011).

National Research Council (U.S.). 2000. Committee on Enhancing the Internet for Health and Biomedical Applications: Technical Requirements and Implementation Strategies. *Networking Health: Prescriptions for the Internet*. Washington, DC: National Academy Press.

Nugent, R. 2008. Chronic diseases in developing countries: Health and economic burdens. *N Y Acad Sci* 1136:70–9.

O'Connor, A. M. 1999. Consumer/patient decision support in the new millennium: Where should our research take us? *Can J Nursing Res* 30(4):257–61.

O'Connor, A. M., A. Rostom, V. Fiset, J. Tetroe, V. Entwistle, H. A. Llewellyn-Thomas, M. Homes-Rovner, M. Barry, and J. Jones. 1999. Decision aids for patients facing health treatment or screening decisions: Systematic review. *Br Med J* 18:731–4.

O'Connor, A. M., D. Stacey, V. Entwistle, H. A. Llewellyn-Thomas, D. Rovner, M. Holmes-Rovner, V. Tait et al. 2003. *The Cochrane Database of Systematic Reviews: Decision Aids for People Facing Health Treatment or Screening Decisions*. The Cochrane Library. http://gateway1.ovid.com:80/ovidweb.cgi. (accessed March 14, 2011).

Office of the National Coordinator for Health Information Technology. 2011. http://healthit.hhs.gov/portal/server.pt/community/healthit_hhs_gov__hitech_programs/1487 (accessed March 23, 2011).

Orman, L. 1983. Information independent evaluation of information systems. *Inf Manag*e 6:309–16.

Osheroff, J. A., J. M. Teich, B. Middleton, E. B. Steen, A. Wright, and D. E. Detmer. 2007. A roadmap for national action on clinical decision support. *J Am Med Inf Assoc* 14(2):141–5. doi:10.1197/jamia.M2334.

Overhage, J. M., W. M. Tierney, X.-H. Zhou, and C. J. McDonald. 1997. A randomized trial of "corollary orders" to prevent errors of omission. *J Am Med Inform Assoc* 4(5):364–75.

Patel, V. L., J. F. Arocha, and D. R. Kaufman. 2001. A primer on aspects of cognition for medical informatics. *JAMIA* 8(4):324–43.

Patel, V. L., A. W. Kushniruk, S. Yang, and J. F. Yale. 2000. Impact of a computer-based patient record system on data collection, knowledge organization, and reasoning. *JAMIA* 7(6):569–85.

Patrick, K., and S. Koss. 1995. Consumer health information "white paper"(draft). Consumer Health Education Subgroup. Committee on Applications and Technology. Information Infrastructure Task Force.

Patrick, K., and S. Koss. 1995. Consumer health information "White Paper." Consumer Health Information Subgroup, Health Information and Application Working Group, Committee on Applications and Technology, Information Infrastructure Task Force. Working Draft.

Payne, J. W. 1976. Task complexity and contingent processing in decision making: An information search and protocol analysis. *Organ Behav Hum Perform* 16:366–87.

Pedersen, C. A., and K. F. Gumpper. 2008. ASHP national survey on informatics: assessment of the adoption and use of pharmacy informatics in U.S. hospitals, 2007. *Am J Health Syst Pharm* 65:2244–64.

Potts, A. L., F. E. Barr, D. F. Gregory, L. Wright, and N. R. Patel. 2004. Computerized physician order entry and medication errors in a pediatric critical care unit. *Pediatrics* 113(1):59–63.

Ralston, J. D., D. Carrell, R. Reid, M. Anderson, M. Moran, and J. Hereford. 2007. Patient web services integrated with a shared medical record: Patient use and satisfaction. *J Am Med Inform Assoc* 14(6):798–806.

Robinson, T. N., K. Patrick, T. R. Eng, and D. Gustafson. 1998. An evidence-based approach to interactive health communication: A challenge to medicine in the information age. *Journal of the American Medical Association* 280(14):264–1269. doi: 10.1001/jama.280.14.1264

Riggio, J. M., R. Sorokin, E. D. Moxey, P. Mather, S. Gould, and G. C. Kane. 2009. Effectiveness of a clinical-decision-support system in improving compliance with cardiac-care quality measures and supporting resident training. *Acad Med* 84(12):1719–26.

Rosenberg, S. N., T. L. Shnaiden, A. A. Wegh, and I. A. Juster. 2008. Supporting the patient's role in guideline compliance: A controlled study. *Am J Manag Care* 14(11):737–44.

Sainfort, F., and B. C. Booske. 1996. Role of information in consumer selection of health plans. *Health Care Financ Rev* 18(1):31–54.

Schkade, D. A., and D. N. Kleinmuntz. 1994. Information displays and choice processes: Differential effects of organization, form, and sequence. *Organ Behav Hum Decis Processes* 57:319–37.

Schoen, C., K. Davis, R. Osborn, and R. Blendon. 2000. *Commonwealth Fund 2000 International Health Policy Survey of Physicians' Perspectives on Quality*. New York: Commonwealth Fund.

Scott, G. C., and L. A. Lenert. 1998. *Extending Contemporary Decision Support System Designs to Patient-Oriented Systems*. Orlando, FL: Paper presented at the American Medical Informatics Association.

Shackel, B. 1991. Usability—context, framework, definition, design and evaluation. In *Human Factors for Informatics Usability*, ed. B. Shackel and S. Richardson, 376–80. Cambridge, UK: Cambridge University Press.

Silberg, W. M., G. D. Lundberg, and R. A. Musacchio. 1997. Assessing, controlling, and assuring the quality of medical information on the Internet: Caveant lector et viewor—let the reader and viewer beware. *JAMA* 277(15):1244–5.

Shim, J. P., M. Warkentin, J. F. Courtney, D. J. Power, R. Sharda, and C. Carlsson. 2002. Past, present, and future of decision support technology. *Decis Support Syst* 33:111–26.

Shu, K., D. Boyle, C. D. Spurr, J. Horsky, H. Heiman, and P. O'Connor et al. 2001. Comparison of time spent writing orders on paper with computerized physician order entry. *Medinfo* 10(Pt 2):1207–11.

Slocum, J. W., and D. Hellriegel. 1983. A look at how manager's minds work. *Bus Horiz* 26:58–68.

Slovic, P. 1995. The construction of preference. *Am Psychologist* 50(5):364–71.

Sneha, S., and U. Varshney. 2009. Enabling ubiquitous patient monitoring: Model, decision protocols, opportunities and challenges. *Decis Support Syst* 46(3):606–19. doi: 10.1016/j.dss.2008.11.014.

Sonnenberg, F. A. 1997. Health information on the Internet: Opportunities and pitfalls. *Arch Intern Med* 157:151–2.

Stacey, D., C. L. Bennett, M. J. Barry, N. F. Col, K. B. Eden, M. Holmes-Rovner, H. Llewellyn-Thomas, A. Lyddiatt, F. Légaré, and R. Thomson. 2011. Decision aids for people facing health treatment or screening decisions. *Cochrane Database of Systematic Reviews* 10(CD001431). doi: 10.1002/14651858.CD001431.pub3

The National Committee for Quality Assurance. 2011. Health plan report card. http://reportcard.ncqa.org/plan/external/Plansearch.aspx (accessed March 19, 2011).

Todd, P., and I. Benbasat. 2001. An experimental investigation of the impact of computer based decision aids on decision making strategies. *Inf Syst Res* 2(2):87–115.

Togo, D. F., and J. N. Hood. 1992. Quantitative information presentation and gender: An interaction effect. *J Gen Psychology* 119(2):161–7.

Tu, H., and G. Cohen, 2008. Tracking Report No. 20. HSC Home Page. http://hschange. org/CONTENT/1006/?PRINT=1. Retreived 4 November 2011.

Turisco, F., and J. Case. 2001. *Wireless and Mobile Computing*. Oakland, CA: California Healthcare Foundation.

United States Department of Health and Human Services. 2011. Breaches Affecting 500 or More Individuals. http://www.hhs.gov/ocr/privacy/hipaa/administrative/breachnotificationrule/breachtool.html (accessed March 14, 2011).

United States Department of Health and Human Services, Office of the National Coordinator for Health Information Technology, HIT Policy Committee: Request for Comment Regarding the Stage 2 Definition of Meaningful Use of Electronic Health Records (EHRs). 2011. http://healthit.hhs.gov/portal/server.pt/gateway/PTARGS_0_0_5383_1472_17094_43/http;/wci-pubcontent/publish/onc/public_communities/u_z/wg_month_pages/mu_jan_portlet/files/nr_mu_rfc__v_4__2011_01_05.pdf (accessed March 23, 2011).

Wagner, E. H., B. T. Austin, and M. Von Korff. 1996. Organizing care for patients with chronic illness. *Milbank Q* 74(4):511–42.

Wager, K. A., F. W. Lee, A. W. White, D. M. Ward, and S. M. Ornstein. 2000. Impact of an electronic medical record system on community-based primary care practices. *J Am Board Family Pract* 13:338–48.

Ward, R., C. Stevens, P. Brentnall, and J. Briddon, 2008. The attitudes of health care staff to information technology: A comprehensive review of the research literature. *Health Information & Libraries Journal* 25(2):81–97. doi:10.1111/j.1471-1842.2008.00777.x

Weingart, S. N., B. Simchowitz, L. Shiman, D. Brouillard, A. Cyrulik, R. B. Davis, T. Isaac et al. 2009. Clinicians' assessments of electronic medication safety alerts in ambulatory care. *Arch Intern Med* 169(17):1627–32.

Williams, A., D. Herman, J. Moriarty, T. Beebe, S. Bruggeman, E. Klavetter et al. 2008. HIPAA Costs and patient perceptions of privacy safeguards at mayo clinic. *Joint Commission J Qual Patient Saf* 34(1):27–35. http://www.ingentaconnect.com/content/jcaho/jcjqs/2008/00000034/00000001/art00005 (accessed March 14, 2011).

Workgroup on Electronic Data Interchange. 2000. HIPAA: Changing the health care landscape. *Oncol Issues* 15(4):21–3.

World Health Organization. 2000. The World Health Report 2000: Health systems: improving performance. http://www.who.int/whr/2000/index.htm (accessed April 11, 2002).

Wyatt, J. C. 1997. Commentary: Measuring quality and impact of the World Wide Web. *Br Med J* 314(7098):1879–81.

Zeng-Treitler, Q., S. Goryachev, H. Kim, A. Keselman, and D. Rosendale. 2007. Making texts in electronic health records comprehensible to consumers: A prototype translator. *AMIA Annu Symp Proc* 846–50. Annual Symposium Proceedings/AMIA Symposium.

31 Why We Play

Affect and the Fun of Games—Designing Emotions for Games, Entertainment Interfaces, and Interactive Products

Nicole Lazzaro

CONTENTS

31.1 WITHOUT EMOTION THERE IS NO GAME

Shakespeare designed the emotional space between characters; game developers design the emotional space between players and game.

Researchers have only just begun to explore the role emotion plays in human activities. In designing emotional responses, most aesthetic disciplines treat audiences as consumers of content and pay little attention to designing emotions from interaction or contribution. Whether it is a movie, a bottle of perfume, or a website advertisement, a broadcast model is used to elicit emotions from observers rather than participants. Scientific research is now beginning to show how emotion influences cognition and behavior offering new opportunities to solicit emotion through action. Emotion emerges from and plays a part in most activities, from following a goal to just goofing around (Damasio 1994; Ekman 2003; Norman 2004; Lazzaro 2004b). Video games lead the way as interactive products that create emotion. More emotional than software and more interactive than films, games manipulate player affect to create poignant experiences. How they do this provides lessons for the design of games, entertainment interfaces, and other interactive products.

Emotion is essential to maintain player focus, make decisions, perform, learn, and enjoy the process of play. Emotion-rich stimuli grab players' attention, such as a swashbuckling adventure in *Sid Meier's Pirates!* The emotions surrounding swordplay increase players' immersion and negative affect or mindset focuses players on applying effort to overcome obstacles. Meanwhile, positive affect from finding pirate treasure improves exploration of alternatives. Exploring options to get the player's sticky ball up on top of a table in *Katamari Damacy* is made easier by the positive affect created as it squashes and picks up cutely rendered bon-bons, toys, and other items. Strong emotional states also allow easier formation and recollection of memories, especially if the user's emotion matches the emotion of the item to be remembered (Ekman 2003). Special moves in *Top Spin Tennis* that offer an emotional response in an opponent are easier to remember and motivate the search for more. Games are innovators in the design of emotional responses integrated into the activity to accelerate it or provide friction for game goals. Each affective state, each emotion, carves a unique signature into a player's psychology, physiology, and behavioral state to create a player experience. Hoping that graphic realism alone will create emotions is nearly the same as adding more background color to fix a usability issue.

Lessons about game experiences inform entertainment interfaces and other product experiences. Games are not just entertaining; they are self-motivating activities. User-experience designers for all types of products can take advantage of how games create these emotions from participation. To transition from interface design to user experience design, interfaces need to be more than "transparent." In games, the interface that makes everything easy robs the player of the fun. Pushing a button for a car to drive itself is less thrilling than winning the Grand Prix. In software, the interface cannot do it all because it does not have the user's knowledge. Therefore, not only must interfaces get out of a user's way, they should allow the user to express him or herself by motivating and supporting the cognitive and behavioral functions required for use. Designers can take inspiration from games to fashion emotional responses during interaction; for instance, computers can detect and respond to user emotion, such as "being more helpful" when software detects frustration in a user's face or fingertips. Like interchangeable colored lenses, games employ emotion-producing attributes to support human performance: rose-colored here for a mood boost, yellow for sharper vision later on, green night vision with a distance indicator for the dark corners, and black to win style points from others. Current usability methods (increasing efficiency, effectiveness, and satisfaction) mostly remove frustration points; they do not yet include techniques to measure and craft other emotions. To exaggerate, a 100% usable product would be boring once it eliminates all the challenge. Customers strap software onto their boots like a crampon, but it should not do the job so well that it climbs the mountain for them (Lazzaro 2004a). It is not a productivity tool if one button-click creates the whole spreadsheet. User experiences should focus on making the process of the task not only easier, but also more enjoyable.

31.1.1 Forget Usability: Make It Fun

> "I don't want to feel like the game just wasted two hours of my life."
>
> **—A *Might and Magic* player**

Wasting time has a whole different meaning to a gamer. Unlike a spreadsheet, the outcome of a game is in the experience of play rather than in the quality of the end result and so is harder to quantify. Traditionally, productivity focuses more on designing a process that creates a better end-product or result. Game design, on the other hand, focuses mostly on designing interactive play that is enjoyable in its own right. Rather than efficiency, game enjoyment requires the pure pleasure of the experience and a fair degree of frustration (Lazzaro 2004a). Similar to user experiences, player experiences are created when what happens in the game affects the player internally as well as externally. Player experiences are

Productivity Software Goals	Game Goal
Task completion	Entertainment
Eliminate errors	Fun to beat obstacles
External reward	Intrinsic reward
Outcome-based rewards	Process is its own reward
Intuitive	New things to learn
Reduce workload	Increase workload
Assumes technology needs to be humanized	Assumes humans need to be challenged

FIGURE 31.1 The goals of productivity and game experiences have several important differences. (From Lazzaro, N., and K. Keeker. *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction Conference (CHI)*, Vienna, Austria, 2004. With permission.)

the combination of emotion, thoughts, and other sensations that occur inside and in between players during play. Beyond usability, player experiences design focuses on affect as well as ease of use (Figure 31.1).

The interactive entertainment offered by games provides unique opportunities to create emotions in the player and unique challenges for the design professional. A game must be usable enough to play but not so usable as to allow someone to push a button and win. Players crave the illusion of superior control that allows them to accomplish more than others including out-thinking the game designer, but not so much control that they lose their way and do not have a good time.

31.1.2 Emotions Are for More than Entertainment

> "Experience is the feeling of what happens."
>
> **—Damasio 1994**

Playing games in their discretionary time, gamers mainly play for the emotions the games create. Recent neuropsychology research suggests that two interconnected information-processing systems continually scan the environment to create a person's experience of the world. A person's *cognitive system* interprets and represents the world internally in order to reason, understand, and interact with it. A person's *affective system* interprets external and internal stimuli relative to goals and needs. This affective system kicks in with an emotional and physiological reaction before a cognitive response is ready. Ideas, thoughts, memories, and knowledge are components of cognition; emotions, moods, sentiments, and other internal sensations comprise a person's affective response (Damasio 1994; Norman 2004). From a cognitive-psychology perspective, cognition understands the world and affect evaluates it (Norman). On a basic level, we bring items with positive affect (sweet-tasting, soothing voices, warm to the touch) closer to us. We push (generally speaking) objects with negative affect (bitter, bloody, sharp, diseased) away from us. In the context of games, the study of affect must also include the discussion of enjoyment.

Player experiences emerge inside the player from the process of interacting with the game. Player-experience design crafts these cognitive and affective responses in conjunction with user behavior. Therefore, the design of player experiences must refine not only the gamer's cognitive response to a system, for example, by reducing complexity; it must also design the gamer's affective response, for example, by inspiring interest or rewarding success in order to increase engagement and support cognitive tasks. For the purposes of game design, affect supports cognitive as well as behavioral tasks, because emotion has a significant effect on enjoyment, attention, memory, learning, and performance.

Emotion and cognition walk hand in hand. According to Ekman, beyond entertainment, emotions are about goals and the things we care about (2003). This makes the creation of emotions ideally suited for game design because most gameplay offers goals with levels and scores to indicate progress. In films, we feel emotions only if we somehow identify with the characters on the screen and vicariously feel their emotions (Boorstin 1990). This also happens in games, but more central to interactive entertainment is when players feel emotions from what they accomplish and fail at during the game. In productivity, feedback on effectiveness, such as creating a sales presentation, happens after the task of creating it is complete. In games, the success-feedback loop is immediate and built into the process of play.

31.2 REQUIREMENTS FOR A PLAYER EXPERIENCE FRAMEWORK

31.2.1 Emotion's Five Impacts on Play

> "Emotions prepare us to deal with important events without us having to think about what to do."
>
> **—Ekman 2003, p. 20**

Games entertain with emotions so players enjoy the ride.

Emotions impact player experiences in five ways. What players like most about games is not the packaging, graphics, or the artificial intelligence (AI), but the total experience that the game creates for the player. This experience lies in the cognitive, affective, and behavioral changes gamers create for themselves as they play. Emotion generates a big part of the entertainment value, the opportunity for challenge and mastery, the thrill from novelty, the ticket to relaxation, and the opportunity to hang out with friends. Customers buy this ride designed by game designers. These emotions create moving gameplay and make a victory taste sweet (Figure 31.2).

Not only does the act of play produce emotions, but also emotions and affect substantially influence the player and how he or she plays. Unlike user experiences, the primary aim of player experiences is to move the player emotionally along with or counter to the game goal. Such techniques heighten the emotional response in the player. Emotions entertain players, focus their attention, help them decide, aid their performance, and assist and motivate learning. Enjoyable emotions from gameplay increase motivation to play further in order to experience more emotions, which makes games self-motivational. Emotions are there from the first click to final volley.

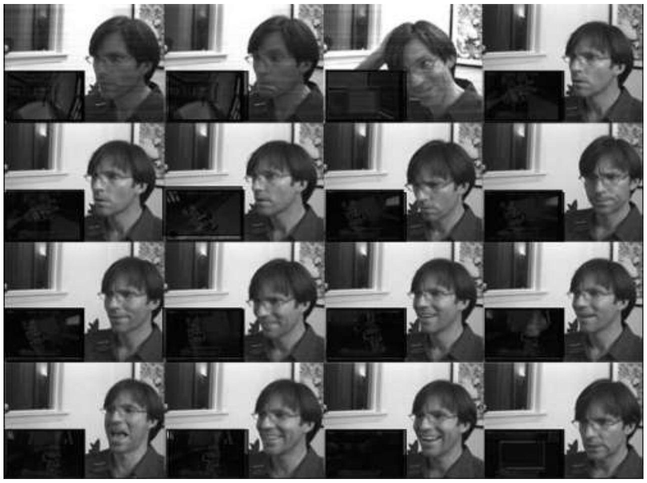


FIGURE 31.2 Game interaction involves and creates many emotions that are clearly visible on the face during play, as seen here during the tutorial for a popular action game. (Courtesy of XEO Design. Copyright © 2004 XEODesign, Inc. All rights reserved.)

31.2.2 Emotion during Play Helps Gamers

1. *Enjoy:* Creates entertainment from strong shifts in internal sensations
2. *Focus:* Directs effort and attention
3. *Decide:* Aids decision making
4. *Perform:* Supports different approaches to action and execution
5. *Learn:* Provides motivation for learning, aids in memory, and rewards progress

Games heighten emotional responses to increase *enjoyment*. *Mario Kart*, for example, adjusts speed of players to keep them together for "close" races; likewise, *Jak and Daxter* keeps players just ahead of rolling boulders. A choice between certain death and escape via a narrow window ledge in *Tom Clancy's Splinter Cell* provides *focus* to fill attention in a way that scrolling through options in a word processor's drop-down menu or walking down an office corridor of options does not. Strong exaggerated affect aids player *decisions* whether it is to attack a goblin in *Worlds of Warcraft* (*WOW*) or grant a Sim bathroom privileges in *The Sims*. Gameplay creates moods that aid players to perform; negative affect during a fire fight in *Battlefield* improves pursuing a narrow course of alternatives such as stopping a sniper, while positive affect from munching letter sounds in *Bookworm* improves identification of new word patterns in this *Scrabble*-like tile game. A player *learns* to keep diners happy in *Diner Dash* when an angry customer empties the tip jar.

1. *Enjoy:* Emotions create strong shifts in internal sensations to heighten and refresh player experiences.

Suspend players over boiling lava or confront them with hideous boss monsters and the game heightens their emotions. Players look for emotional rewards as well as a high score. Much of the enjoyment comes from the player's affective response and ultimately separates player experiences from user experiences. In games, player participation is essential. The emotions come from the players' efforts in accomplishing a task and game enjoyment centers on the experience of strong emotional shifts from their actions. Games provide the environment where the player becomes the central hero to accomplish the extraordinary. The buddy rescued or the enemy vanquished in *Battlefield* is no more real than in film, but the player's role in the achievement is. An interactive medium, the choices offered by the game (including the graphical user interface) must sharply enhance the experience. A poor interface reduces interactivity and harms the game experience. Therefore, a large part of good game design lies in creating effective interfaces that also create emotional responses. Movies, by comparison, invite the audience to share in the joys and sorrows of characters on screen. Where games move beyond film to claim their true power is by rewarding player action with emotions. Movies can never hand the audience a jet ski for the thrill of stopping the impending global thermonuclear war. As part of their unique value proposition, games have to

2. *Focus:* Emotion directs effort and attention aids or influences many aspects of cognition by focusing attention, providing immediate feedback and rewards.

Emotion supports cognitive tasks by directing, focusing, and holding attention, creating absorbing engagement, and at the same time allowing emotion-laden ideas to dominate thought (Clore and Gasper 2000 as cited in Brave and Nass 2003). Events are filtered through mood, we attend more to thoughts that match the current mood, and ideas similar to a user's current mood are remembered better (Ekman 2003; Thorson and Friestaad 1985 as cited in Brave and Nass 2003). Emotional stimuli rewards more detailed analysis.

In games, tight feedback loops reward player actions with immediate visual and audio feedback that motivate the player to want to take another action. This rewarding-stimulus-response loop is a powerful motivating force that reaches a wide mass-market audience. Several game-design techniques magnify the effect emotions already have to focus players on a task. For example, game obstacles sometimes use negative affect to increase player focus and, at other times, they use positive affect to increase creativity and problem solving to provide interesting behaviors or situations to encourage players to explore (using positive affect to increase creativity and problem solving). Emotions reward attention. Game feedback provides new stimuli to interpret and creates new experience. Rewards create positive and negative affect. Failures create negative affect with the hopes that players will double their efforts and try again.

3. *Decide:* Emotion is vital to decision making in games.

Humans use both thought and emotion to make decisions. Experimental evidence suggests that people with damage to brain structures involved in emotions can generate appropriate logical options and discuss each decision's impacts and tradeoffs with great clarity, but are unable to make the actual choice itself (Damasio 1994; Norman 2004). When we select an entrée from a restaurant menu, it is what we "feel" like having. There are also logical components to that choice (calories, fiber, who's paying, etc.), but there is something other than logic that happens inside that helps us choose and "feel" good about the selection. Some people even feel that their car drives better after it has been washed or had the oil changed (positive affect). Emotions also let people make snap decisions (Ekman 2003; Norman 2004). Emotions help players decide and combine with logic to make these decisions interesting. Because positive and negative affect guides players, it is easier to choose between options with strong emotional stimuli. Game interfaces that supply strong emotional responses also have this effect. Games also use affect to add conflict. Either the emotions support the objective or make achieving it more difficult as players resist the urge to run away. Recognizing this initial role emotion plays in decision making offers a crucial way to improve decision-support products.

4. *Perform:* Emotion supports different approaches to action and execution.

Emotions are a key component in most tasks. This is easy to see in games, which are often designed to create a particular affective state, which sometimes requires following strict detailed procedures with zero tolerance for error or creative exploration of alternative options. The lesson for human–computer interaction (HCI) is that properly designed emotions support the right affective state to get the job done as well as increase appeal. In productivity software, this is important for both the task (such as struggling to find the right word while writing) and the software (such as struggling to find the right feature), and both create frustration. Like a car's seatbelt alarm, certain levels of frustration create mild negative affect in the user and direct the user toward a certain action. Too much negative affect (such as when players cannot keep cars on the racetrack) makes players feel like they will fail and quit. Software needs aspects that create emotions to support tasks, and allow options that let users balance their own levels of frustration. Like cycles of hard and easy things to do in a game, users need to experience new emotions to refresh their experiences. Emotion creates moods that persist and help the player perform. The emotion the designer chooses should help the actions required. Negative affect narrows attention on aspects relevant to the problem, while positive affect opens it to explore new alternatives. Positive moods influence creativity and flexible thinking for problem solving. Relaxed and happy thought processes expand and become more creative and imaginative, and make players more tolerant of minor difficulties (Norman 2004). Moods with a negative affect focus attention, while positive moods help players generate new options. Cycling between moods creates variety to refresh the experience, and offers the option of approaching a problem with a different problem-solving strategy related to the new mood.

5. *Learn:* Emotion provides motivation and rewards progress.

Played for pure enjoyment, game emotion motivates players to pay closer attention and repeat an action, enabling them to master highly complex interfaces and interactions, learn countless features and strategies, and spend hours doing this even though they may fail repeatedly. To increase product mastery, many game methods apply nongame interfaces. While productivity software users prefer to learn the bare minimum number of features to accomplish their work rather than achieving level 42 in spreadsheet wizardryness, game-like motivation may increase exploration of additional product features. All games involve learning. Emotions reward changes and growth inside players as they master what they could not do before. Learning and remembering are easier for emotional stimuli, and when experiencing an emotion or a mood, it is easier to remember thoughts that have a matching emotional context. Improved performance such as learning a new skill or high score is a big part of the enjoyment of play.

31.2.3 Requirements for an Emotion Framework for Games

> "The problem with the words 'enjoyment' and 'happiness' is that they're not specific enough; they imply a single state of mind and feeling, in the same way that the terms 'upset' and 'negative' don't reveal whether someone is sad, angry, or disgusted."
>
> **—Ekman 2003, p. 190**

A practical methodology for designing emotions should support designers as well as researchers to build better player experiences. More than a model, a practical method for designing and examining player experiences should take into account four perspectives: (1) what players like most, (2) what creates emotions, (3) what game designers can control, and (4) what researchers can measure. The method's components must be observable, salient to the player, relevant to the player's experience of fun, and apply to a wide variety of game genres and hardware platforms. It must account for how activities offered by best-selling games make them more popular than others with similar features. It should track internal and external aspects of play including complex emotions and their impact on players' lives. A framework must capture emotions' role in enjoyment, maintaining player interest, decision making, performance, and learning. Designing for emotion in games should build on the research and theories of several disciplines. It must cover differences in expectations and play patterns such as Bartle who first classified players as (a) Achievers, (b) Explorers, (c) Socializers, or (d) Player Killers (Bartle 1996). It must measure emotion from play from different game components such as challenge and fantasy (Malone 1981) and biometrics such as heart rate, control pressure, and facial expression (Mandryk 2004; Sykes and Brown 2003; Hazlett 2003), and specific emotional states such as pleasure and aggression using FMRI (Weber, Ritterfeld, and Mathiak 2006). It should

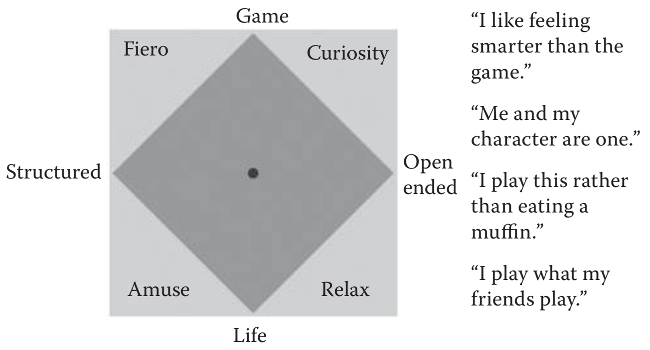

Game
Fiero
Curiosity
Structured
Open ended
Amuse
Relax
Life
"I like feeling smarter than the game."
"Me and my character are one."
"I play this rather than eating a muffin."
"I play what my friends play."

FIGURE 31.3 Players play for "the ride" that games take them on. Player experiences (PX) generate emotions from reactions along two axes: goal-directed versus open-ended play, what happens inside the game versus how gameplay relates to other aspects of the player's life. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

cover emotions from game goals, open-ended play, as well as emotions from game events and those that happen in real life. Most importantly, the method must describe which gameplay mechanisms produce specific emotions that are central components of the player experience (Figure 31.3).

31.2.3.1 Evaluating Relevant Frameworks for Emotion, Products, and Entertainment

A limitation shared by current models of emotions and products is that they focus more on positive and negative affect than on how to create specific emotions from interaction; and even less attention is given to how the actions of a game player create specific emotions. To be useful to game and product designers, player-experience methodologies must identify more emotions than simply positive and negative affect. Affective states grow and change during gameplay.

Many emotions are enjoyable, such as curiosity, which can also lead to other positive emotions such as wonder and even love. However, players also enjoy play experiences containing negative emotions. The pleasure that comes from many game designs requires mastering difficult situations or experiencing unpleasant emotions. How emotions grow and change during play is a critical part of player experiences. During game testing, designers need more information than whether an action was or was not fun. Player observation leads to the design of better games, but this requires a deeper understanding of the player experience and better techniques on how to measure it (Figure 31.4).

31.2.3.2 Flow the Optimal Experience: Csikszentmihalyi

The most relevant and influential psychological research outside of games is Csikszentmihalyi's model of on optimal experiences or "flow" (Csikszentmihalyi 1990). He found that people are happiest when engaged in intrinsically motivating activities such as rock climbing, dancing, and gardening. These activities offer long-term, deep, memorable experiences that require concentration and growth. He noted that optimal experiences carefully balance skill with difficulty to create a deeply enjoyable mental and physical state. Without challenge, an activity becomes routine and boredom sets in. With too much challenge, the person becomes too anxious and leaves because he or she feels frustrated (Csikszentmihalyi 1990). The activity should aim to achieve flow through experience design and create forward movement and personal development through pleasurable engaging experiences. With its balance of difficulty and user skill, flow clearly describes a critical component of play experiences. However, Csikszentmihalyi's model only took into account two emotions: (1) anxiety (frustration) and (2) boredom, and ignored several other emotions essential for gameplay. While these first two emotions are important in games, many other emotions play an important role in player engagement. Csikszentmihalyi grouped together physical, mental, and aesthetic challenges, which prevented discussing emotional effects of art and audio separately from the challenge of gameplay. A model for emotions in games should connect the emotions most important to games to how games create them.

31.2.3.3 Pleasure from Products: Jordan, Norman, and Boorstin

Pleasure comes from different aspects of experiencing a product. To define what he calls the new human factors, Jordan expanded on anthropologist Tiger's Four Pleasures model to create a framework for thinking about pleasure from products. Jordan discussed (a) the ideo-pleasure, (b) physio-pleasure, (c) psycho-pleasure, and (d) socio-pleasures of a product (or emotions from a product's idea, physical, psychological, and social attributes). Each of these Four Pleasures describes enjoyment from a different perspective including how the use, ownership, or identification with a product produces emotion (Tiger 1992, as cited in Jordan 2000; Jordan 2000). Changing an aspect can strengthen a desired emotional response. Norman built on and simplified Jordan's model to three layers of mental processing: (1) the Visceral (an automatic biological response), (2) the Behavioral (learned actions), and (3) the Reflective (involving thought, self-image and relationship to others) (Norman 2004). Interestingly, filmmaker Jon Boorstin also proposed three ways that films delight audiences: (1) the Visceral Eye (enjoyment on the biological level such as automatic sensory reactions to explosions or speed), (2) Voyeuristic Eye (enjoyment through seeing events unfold), and (3) the Vicarious Eye (enjoyment through identification with people on screen) (Boorstin 1990). Unfortunately, none of these frameworks go deeply into what actions create specific emotions.

31.2.3.4 Enter Ekman and Facial Expressions

Through his research of universal facial gestures, Paul Ekman's Facial Action Coding System (or FACS) and a compilation of others' research, Ekman's work has increased understanding of specific human emotions' cross-cultural boundaries. His fascinating and highly accessible book *Emotions Revealed* described how to identify emotions through facial gestures and the important roles emotions play in our lives (Ekman 2003) (Figure 31.5).

Comparison of Models and Methods to Create Affect from Products and Entertainment Experiences

XEODesign Four Fun Keys Model	Hard Fun Fiero	Easy Fun Curiosity	Serious Fun Relaxation	People Fun Amusement
	Challenge Game, Goal	Novelty, Fantasy Game, Open Ended	Real World Purpose Life, Open Ended	Social Life, Goal
Bartle's Original 4 Player Types (1996, 2003a, 2003b)	Achiever Player Killer	Explorer		Socializer Player Killer
Boorstin (1990)		Voyeuristic Eye Visceral Eye		Vicarious Eye
Csikszentmihalyi (1990)	Enjoyment, flow	Pleasure, microflow		
Ekman (2003)		Auto appraisal, memory of emotion, imagination, reflective appraisal		Empathy with another, Violation of social norm, talking about emotion, making facial expression of emotion
Hassenzahl et al. (2000)	Ergonomic quality	Hedonic quality		Community
Kim (2000)				
LeBlanc, Hunicke, and Zubek (2004)	Mechanics, dynamics Aesthetics	Aesthetics		
Malone (1981)	Challenge	Curiosity Fantasy		
Norman (2004)	Behavioral		Reflective visceral	Reflective
Piaget (1962)	Formal games with rules	Sensory-motor play Pretend play		
Tiger (1992)		Physio-pleasure	Ideo-pleasure	Socio-pleasure
Jordan (2000)		Psycho-pleasure		
Wright, McCarthy, and Meekison (2003)	Spatial-temporal thread	Compositional thread Sensual thread Emotional thread		
Common Drama and Theater Constructs	Character "motivation," plot points, objectives 3-act structure	Setting, plot, story, character, suspension of disbelief	Catharsis, music, set and costume design	Character Dialog Acting

FIGURE 31.4 Several frameworks describe the emotion resulting from entertainment experience or the use of a product. Comparing their similarities and differences provides interesting insight into the basic requirements for entertainment and product emotions. (Courtesy of XEODesign. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

Six Plus One Universal Emotions with Universal Facial Gestures

Emotion	Example
Frustration:	Figuring out how to get character off a roof in *Tom Clancy's Splinter Cell* (and all-too-often created by usability issues that detract from the player experience)
Fear:	Falling into boiling lava, fast-moving projectiles aimed at the player in *Doom*
Surprise:	Using *Myst's* linking books for the first time to transport to a new world
Sadness:	When the young magician *Aerith*, in *Final Fantasy* VII, is murdered
Amusement:	When two Sims get married in *The Sims*, or rolling over and picking up sumo wrestlers in *Katamari Damacy*
Disgust:	Becoming a social outcast (social disgust) after losing the dancing challenge in *Sid Meier's Pirates!*
Curiosity:*	Wanting to know what happens by driving the race track the wrong way in *Project Gotham Racing 3*

* Not all researchers (including Ekman) considered curiosity a universal emotion with a unique facial gesture. I include it here as a seventh emotion because of its importance in games and ease of observation.

FIGURE 31.5 Researchers generally agree that there are at least six emotions with universal facial gestures.

Because of its specificity, the FACS coding system for facial gestures offers promise as an emotional measure for games and software, and is the basis for many of XEODesign's observational studies. However, Ekman's work focused on the identification of individuals' experience of specific emotions and stops short of discussing how products or their use can create these emotions. What game designers need is a method that connects specific emotions to player actions in the game.

All of these models lay the groundwork for what is ultimately needed by game designers—a method for producing specific emotions from the interactions players most appreciate during goal oriented and open ended play. To do this the game industry needs an expanded method to create specific captivating emotions from the best-loved types of gameplay. This is a core focus of XEODesign's independent research.

31.3 PLAYER EXPERIENCE FRAMEWORK

31.3.1 Why We Play Games: Four Fun Keys Model

XEODesign conducted independent research to identify four key processes that create emotion in best-selling games (Figure 31.6). Each experience involves a different emotion to create a different Player Experience Profile. By presenting a goal and breaking it into small achievable steps, games create emotions from *Hard Fun*, where the frustration of the attempt is compensated by the feelings of accomplishment and mastery from overcoming obstacles. Outside of goals, games provide novel opportunities for interaction, exploration, and imagination, which create *Easy Fun*. Games that use emotions in play to motivate real-world benefits to help players change how they think, feel, and behave or to accomplish real work create *Serious Fun*. Finally, games that invite friends along get an interpersonal emotional boost from *People Fun* (Lazzaro 2004b). The Four Fun Keys are a collection of related game interactions (game mechanics) that deliver what players like most about games. Each offers a key to "unlock" unique emotions such as frustration, curiosity, relaxation, excitement, and amusement. Best-selling games provide features that support at least three of these Four Fun Keys to create a wider emotional response in the player. To keep things fresh during a single-play session, gamers move between the four different play styles (Lazzaro 2004b). Developing each key focuses and rewards the player with emotion from a self-motivating experience that deepens the game's player-experience profile. Designers of products and productivity software can also use these Four Fun Keys to increase emotional engagement for applications outside of games.

Only some of the emotions from playing basketball in the real world come from the Hard Fun of making baskets. Close examination reveals that all four Fun Keys are part of this popular sport. Dribbling the ball or doing tricks like a Harlem Globetrotter offers Easy Fun from novelty and role play. Intentionally blowing away frustration and getting a workout creates Serious Fun. Competition and teamwork make the game even more emotional from People Fun. All four types of fun make basketball's player experience more enjoyable. None of these require story or character. Through examination of how each type of fun creates emotions, designers and researchers can create better and more emotional player experiences (Figure 31.7).

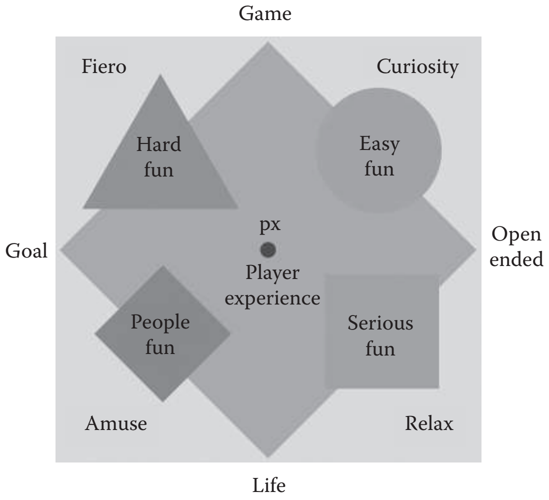


FIGURE 31.6 Each Fun Key is a collection of game mechanics that create a favorite aspect of gameplay. Emotions prepare players for action and reward players. These Four Fun Keys each contribute a unique set of emotions to the game. Changing one of the four key mechanisms will change the Player Experience profile of the game. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

Four Fun Keys to More Emotion through Gameplay

Lead Emotion	Fun Key	Key Game Mechanic
Fiero*	Hard Fun	Affect related to challenge, strategy, and mastery
Curiosity	Easy Fun	Affect related to novelty, ambiguity, detail, fantasy, role-play, and absorbing attention
Relaxation	Serious Fun	Affect related to purposefully changing oneself, learning, or doing real work
Amusement	People Fun	Affect related to competition, cooperation, and socializing with others

* Personal triumph over adversity

FIGURE 31.7 Game mechanics create emotion from what players like most about play. To this group of easily observable emotions, we add frequently reported emotions collected through verbal descriptions of internal sensations as well as player's body language during play. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

31.4 FIERO FROM HARD FUN GAMEPLAY

Gameplay that rewards effort through challenges to create fiero.

31.4.1 Desire for Fiero Helps Players Master Challenge

> "It's easy to tell what games my husband enjoys the most. If he screams 'I hate it. I hate it. I hate it,' then I know two things: a) he will finish it and b) buy version two. If he doesn't say this, he'll put it down after a couple of hours."
>
> **—Wife of a hard-core PC gamer**

The most obvious enjoyment in games comes from mastering a challenge and reaching a goal. Hard Fun is a self-motivating activity that keeps the user focused and enthusiastic by providing an obstacle, an objective, and a score. Hard Fun game mechanics challenge a player to overcome an obstacle to achieve a goal. Hard Fun experiences reward mastery, either explicitly with points and bonuses or implicitly through new levels or abilities. Because this type of play requires application of effort, we call this Hard Fun. In our research, we expand on this phrase (Csikszentmihalyi 1990; Papert 2005) to define Hard Fun as the rewarding process of mastering a challenge that involves the creation and testing of strategies and the application of effort. Hard Fun rewards effort and discourages failed approaches. Hard Fun creates the emotions of frustration and boredom. More importantly, it produces *fiero*, the Italian word Isabella Poggi and Ekman used to describe the personal feeling of triumph over adversity (Poggi, as cited in Ekman 2003). One of the most important game emotions, fiero is a strong feeling of personally accomplishing something difficult such as defeating the boss monster (Figure 31.8).

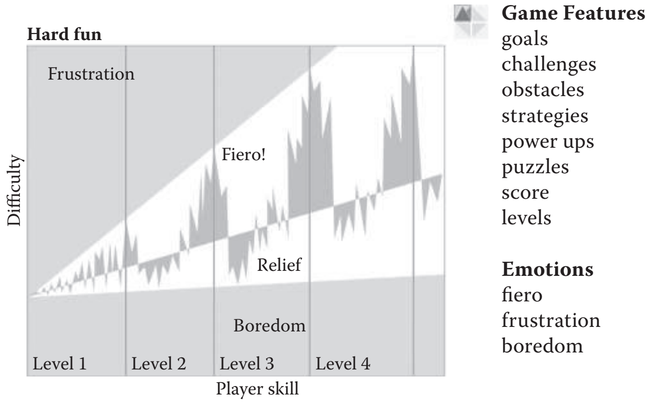


FIGURE 31.8 Similar to Csikszentmihalyi's concept of flow, games must balance difficulty with player skill and provide enough variety of challenges, strategies, and puzzles. If the player gets too frustrated or bored, he or she leaves the game. Charting a player's progress through a game onto Csikszentmihalyi's flow model illustrates that in addition to the requirements for flow, games can be made more emotional from sharp changes in the level of difficulty for game challenges. Game difficulty increases to the end of a level and then resets at the start of the next. When players succeed at the point where they are about to quit, they are more likely to experience fiero. In this figure, fiero occurs at the end of level two before continuing to level three of this game (modified from Csikszentmihalyi 1990). (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

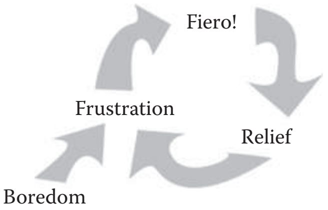
Hard Fun requires a high investment of energy from the player. By perfectly balancing player skill with game difficulty, Hard Fun meets many of the characteristics and requirements for flow (Csikszentmihalyi 1990; Lazzaro 2004b). For example, *Pac Man's* simple game mechanic (eat dots and avoid ghosts) offers clear long-term and short-term goals, the opportunity to concentrate, achievable tasks, an uncertain outcome, and immediate feedback to player decisions. It creates a deep sense of control through tight feedback loops between player input and action in the game. All of these enhance players' absorption into a challenging activity and improve their ability to perform. Beyond flow's balance of difficulty and skill, game designers do other things to change how players feel about their progress in the game. According to game designer Steve Meretzky, rewards along the path to the goal enhance enjoyment such as power ups, big jumps in score, animations and sounds (Lazzaro 2005a). For example, the power ups in *Pac Man* create super-charged feelings as players turn the tables to chase ghosts.

Hard Fun focuses player attention on achieving results by providing opposition and constraints, such as removing an alien threat in *Halo* or aligning puzzle pieces in *Tetris*. Often games provide a choice of strategy, and games with high replay ability offer a choice of goals. For example, *The Sims* offers several winning conditions such as the best-looking house, the most friends, or the most money. The new obstacles, constraints, and tradeoffs from different goals suggest multiple strategies and enhance the challenge. These emotions facilitate achievement of a goal, focus play effort, and reward accomplishment. Negative affect increases focus and concentration (Norman 2004), such as when the player encounters resistance to the goal through failed attempts. Sometimes Hard Fun emotions run counter to what needs to be done. For example, performing well or moving toward something rather than running away is more challenging when surrounded by negative affect. Other Hard Fun emotions, such as fiero from achievement, put the player into a positive affective state by rewarding the player's efforts. Unlike many emotions, fiero does not require an audience (Poggi as cited in Ekman 2003) and is a powerful emotion unavailable in film or novels. Players enjoy these emotions from Hard Fun, including the mental focus provided by frustration that helps players concentrate (Figure 31.9).

While general enjoyment results from emotions that provide motivation, other emotions impact the player's response to the game's stimulus. Decision making is aided or made more challenging by affect, whether it is the positive affect of collecting *BeJeweled*'s colored gemstones or the negative

Chain of Hard Fun Emotions Increases Enjoyment

Goal-directed gameplay creates Hard Fun emotions that focus and reward players for overcoming obstacles.



Hard Fun emotions involve obstacles, strategy and success and help players accomplish a goal by focusing and rewarding effort.

Emotion	Common Themes and Triggers
Frustration	Opposition to an important goal, sudden reversal, feeling of being thwarted, physical restraint. Anger prepares the body to remove an obstacle by force (Ekman 2003).
Fiero* (Italian)	Personal triumph over adversity (Ekman 2003). Overcoming difficult obstacles, players raise their arms over their heads. They do not need to experience anger prior to success, but it does require effort and some frustration.
Boredom**	Repetitive, dull, or tedious tasks (Ekman 2003). Lack of interest in the outcome or in playing. Dispelling boredom is also a major reason to play games.

* Fiero is a positive emotion that has a body rather than a facial gesture. It is not yet known what fiero gestures look like across cultural boundaries.

** Ekman does not list boredom as having a universal facial gesture; however, it is frequently seen in games that lack sufficient challenge.

Source: Lazzaro 2004b.

FIGURE 31.9 Hard Fun Emotion Cycle. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

affect from *Bookworm*'s burning letter tiles (which makes their use a priority and picks up the perceived pace of the game). Emotional states also focus effort and attention on available choices in play. Positive affect, for example, helps to generate new ideas such as what gemstones to match in *BeJeweled*, or negative affect, surrounding an impending invasion in *Civilization*, provides additional focus for decision making on how to respond.

31.4.2 Fiero Enhances a Player's Sense of Progress

"I have to concentrate!"

> **—Traveler in St. Louis airport on what he likes most about *JamDat Bowling* on his mobile phone. Playing removes distractions from his consciousness and engrosses him in a rewarding activity while he waits.**

Players enjoy many other aspects of games with emotions outside of flow's balance of difficulty and skill. To enhance a player's sense of progress as play continues, games offer new tools or abilities along with new obstacles, constraints, and tradeoffs to maintain interest. Levels divide challenge into gradually increasing amounts of difficulty. The difficulty also increases inside each level, with most levels ending in a "boss monster" final challenge similar to a major plot point in a story. Defeating boss monsters or boss puzzles produces fiero, and the start of the next level is often dramatically easier to provide emotional relief. The best-selling game, *Halo*, uses Hard Fun to enhance a player's performance by the way it breaks a variety of challenges into levels, rewards progress, provides power ups, such as new weapons or armor, and adds new vehicles to offer fresh strategy options (Figure 31.10).

In fiero, the ultimate game emotion, the sensations are powerful. It is how players feel when they beat the boss monster or make level 20 after difficult struggles, or when they win a tennis match at Wimbledon. As fists punch the air, a victorious player screams, "Yes! I did it!"

In XEODesign's play lab, fiero appears as a positive upward gesture of the arms or body after succeeding a challenge. Some players jump their characters up and down to express this emotion. (*Source:* Lazzaro, N. 2004b. In *Proceedings of the Game Developers Conference.* San Jose, California. www.xeodesign.com/whyweplaygames.html (accessed December 28, 2005)).

FIGURE 31.10 In fiero a player raises an arm to show her excitement. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

31.4.3 Hard Fun Creates Challenge with Strategies and Puzzles

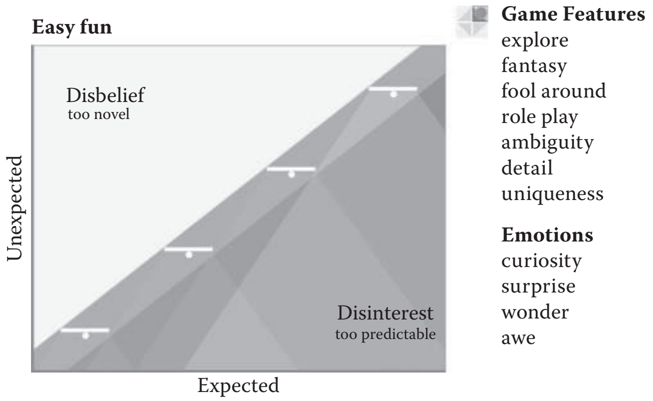
Central to the enjoyment of Hard Fun is the creation and testing of strategies, applying creativity, and the development of skills. While some players focus on the goal, many enjoy the process of learning how to win or take pride in how much better they play with each repetition of the game. Unlike real-world sports such as baseball, computer games increase the challenge by changing the rules and winning strategies between levels to keep the emotions from gameplay fresh. Players must strategize to find new ways to play. Rather than requiring more points in less time, best-selling games such as *Collapse* keep players engaged by making a winning strategy obsolete a few levels later (Lazzaro 2004b). The process of devising new strategies or solving puzzles in new ways is something that players like most about games. The success and failure of these mental tasks and the increase in player skill all create Hard Fun emotion and help players stay engaged and playing.

31.5 CURIOSITY FROM EASY FUN GAMEPLAY

31.5.1 Curiosity Retains Attention

> "Part of the enjoyment comes from the spy technology . . . cool spy tools are part of the Spy experience."
>
> **—Xavier playing *Splinter Cell***

Less apparent but equally important to Hard Fun, top-rated games offer a lot of gameplay outside of or en route to a goal. Easy Fun is a self-motivating activity that maintains player engagement through novelty beyond an obstacle, goal, or score. Easy Fun offers novel interaction to inspire player curiosity to explore, fantasize, and role play. By balancing what is expected and unexpected, careful game design prevents the player from quitting from either disinterest or disbelief. Easy Fun derives from the ability to explore and create exceptional experiences unavailable in the real world. The emotions of curiosity, surprise, wonder, and awe surrounding Easy Fun capture and retain the player's attention, as opposed to Hard Fun where the emotions of frustration in hopes of fiero help players focus on and apply effort toward attaining a goal (Figure 31.11).



Gameplay that fills attention through novelty to inspire curiosity.

FIGURE 31.11 Easy Fun emotions maintain player attention without challenge through novelty and inspiring fantasy. Similar to Csikszentmihalyi's concept of flow, players will leave a game because of disbelief or disinterest. To maintain player interest, the game design must balance the expected with the unexpected. The player experience profile of Easy Fun includes curiosity, surprise, and wonder. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

If Hard Fun revolves around goal achievement, Easy Fun focuses attention by offering opportunities to explore, get lost in a fantasy, role play, or simply horse around. With compelling Easy Fun, the player ignores the goal completely or forgets about keeping score. Easy Fun players find rewarding, open-ended activities on top of the game's main goal. The so-called sandbox play patterns capture player attention and pull players into deep states of immersion through curiosity instead of daring them with challenge. Best-selling games offer opportunities for interaction through unusual yet enjoyable behavior of the controls and novel interaction with the world. Players become fascinated just by interacting with the game, such as being able to flip a car off freeway exit ramps in *GTA* (*Grand Theft Auto*). Like the fiero from Hard Fun, Easy Fun has positive peak emotions that reward play such as surprise, wonder, and that "ah-ha" feeling from figuring something out. However, unlike Hard Fun none of these Easy Fun emotions require frustration. Story can often generate the emotions of wonder and surprise, but role-playing and a player's own discovery through exploration create these emotions on a much more personal level. Positive affect encourages creativity and exploration of alternatives (Norman 2004) that opens the player up to free associations and other possibilities. Easy Fun emotions focus on filling player attention.

Close examination of the conditions, causes, and relationships between emotions provides opportunities for game designers to create even bigger emotional responses in players. It is not that a player cannot feel curious during Hard Fun; but with Easy Fun, curiosity and the sheer joy of interaction drive the player rather than only the score, as it is in Hard Fun. Like improv theater, Easy Fun in games such as *Grand Theft Auto* makes offers to players such as a car and a plate-glass store window. It is up to the player to accept this opportunity and discover how the car and window interact.

31.5.1.1 Chain of Easy Fun Emotions Increases Enjoyment

Open-ended gameplay creates Easy Fun emotions and increases immersion in an activity (Figure 31.12).

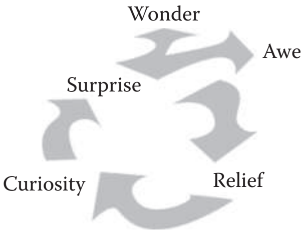
31.5.2 Easy Fun: The Bubble Wrap of Game Design

> "In real life, if a cop pulled me over I'd stop and hand over my driver's license. Here I can run away and see what happens."
>
> **—Xavier playing *GTA Vice City***

Easy Fun is the bubble wrap of game design. It is fun without a purpose. Easy Fun provides novel opportunities for

Chain of Easy Fun Emotions Increases Enjoyment

Unstructured sandbox play creates Easy Fun Emotions that reward players outside of challenge and keeping score



Easy Fun offers emotions surrounding the unique. Games provide a sequence of emotions, often starting with curiosity then creating surprise. Easy Fun can sometimes create wonder or awe if the effect is particularly strong.

Emotion	Common Themes and Triggers
Curiosity*	Unusual, unresolved situation that peaks player's inquisitiveness (Ekman 2003). Something that players find strange, odd, or intriguing, such as *Myst*'s surreal ship-rock island.
Surprise	Sudden change. Briefest of all emotions, does not feel good or bad, after interpreting the event this emotion merges into fear, relief, and so on. (Ekman 2003), such as when matching two blocks clears the whole board.
Wonder	Overwhelming improbability (Ekman 2003). Curious items amaze players at their unusualness, unlikelihood and improbability without breaking out of realm of possibilities, such as in *Myst*'s linking books.
Awe	Combination of wonder with a fear and dread (Ekman 2003), such as a beautiful but impossibly powerful dragon or female warrior in *EverQuest*.

* Not all researchers including Ekman recognized curiosity or interest as a universal emotion with a distinct facial gesture. However, those who did recognize it saw the emotion indicated by a lifting and drawing together of the inner eyebrows. In our research, we saw it frequently combined with leaning forward. It was also a feeling reported verbally by players (Lazzaro 2004b).

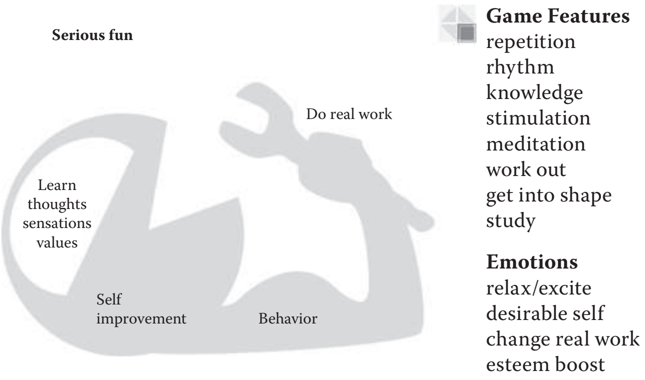
FIGURE 31.12 Easy Fun Emotion Cycle. (Courtesy of XEO Design, Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

interaction that players "discover" alongside of or outside of the main play. The enjoyment and label for Easy Fun comes from the way players goof around, frolic, explore, and play with an ease they do not have when pursuing a specific winning condition as with Hard Fun. Best-selling games such as *GTA, Halo*, and *Myst* offer numerous opportunities for Easy Fun so that when the challenge gets too tough or loses its appeal, the players have many other things to do that create emotional responses. A big role of Easy Fun in best-selling games is to refresh the player between or in the middle of challenges. In *Halo*, for example, players can cycle between the Hard Fun of combat and the Easy Fun of exploring a ring world. The game's battlefield is on the inside of a ring-shaped planet, which inspires curiosity as players investigate as they approach a horizon that instead of dipping down from view, dips up overhead. Easy Fun also provides interest when players pursue the opposite direction of a game goal such as putting the Sims in the pool and removing the ladders in *The Sims* or placing predator and prey animals in the same pen just to see what happens in *Zoo Tycoon* (a flurry of dust and the prey disappears). Through exploring both what's right and what's wrong, games offer more opportunities for emotion, especially from violating norms. In addition to relief from challenge, Easy Fun prevents progress in gameplay from feeling like a skeleton of correct decisions. Easy Fun reinvigorates emotionally and often reinterests the player in the goal. By offering both kinds of fun, the game extends the average play session and lets the player self-regulate the challenge if the Hard Fun becomes too hard.

> "The journey is the reward."
>
> **—Design philosophy at Cyan, creators of *Myst***

The emotions from Easy Fun both inspire and satisfy a player's curiosity. To create the emotions of curiosity, surprise, wonder, and awe in addition to novelty, Easy Fun gameplay uses juxtaposition, where contrast between items or events requires the player to investigate. Like Magritte's surrealist painting, "This is not a pipe," a player of *The Sims* must interpret the Siamese pictograph language spoken by the characters. The player projects in and provides an explanation for any conversation between Sims. The Easy Fun of games provides opportunities for fantasy and role play. Players can take elements of the game and do their own thing with them whether it is wielding an orc's mace in *World of Warcraft (WOW)* or donning a superhero's cape in *City of Heros*. Games also reward player curiosity with details such as in Cyan's *Myst*. In addition to the Hard Fun of puzzles, *Myst* offers Easy Fun gameplay from exploring worlds. To encourage players to slow down and notice, *Myst* provides numerous small details in the environment that reward closer inspection and encourage a slow, careful gameplay style.

Instead of running through at top speed, players spend more time exploring and looking for clues to solve the mystery. This supports the style of interaction needed to play and win the game. In addition to detail, *Myst* captures player attention through ambiguity in the setting, surrealistic juxtapositions such as the boiler room inside a tree, linking books, and faded sketches of other wondrous technology. The conflict that creates emotional tension in Easy Fun is often between what the player knows and does not know. Easy Fun offers detail that rewards player exploration and paying closer attention. Because challenges can feel like a grind, Easy Fun refreshes and provides new and interesting things to do in the game. Easy Fun provides novelty to keep play open ended and interesting not because players wonder about whether they have "the stuff" it takes to reach a goal, but simply to make the player experience interesting, surprising, and far from routine.



Gameplay that changes the self and the real world.

FIGURE 31.13 Serious Fun creates emotions to engage in activities that players hope will change how they think, feel, and behave or that will accomplish real work. The player experience profile of Serious Fun includes relaxation, excitement, boosts in self-esteem, and the satisfaction from a job well done. (Courtesy of XEO Design, Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

31.6 RELAXATION FROM SERIOUS FUN GAMEPLAY

31.6.1 Relaxation Changes How Players Think, Feel, and Behave

"Playing helps me blow off frustration at my boss."

—A hard-core *Halo player*

In Serious Fun, people play games with a purpose (Figure 31.13). They play to improve their lives by changing how they think, feel, or behave. The enjoyment motivates continued engagement with an activity that brings the desired results. The Serious Fun in games reliably relaxes or excites gamers as they play for a purpose to change the player's internal state, develop good habits, improve self-esteem, learn, or do work outside of the game itself. In Serious Fun, the entertainment value captures attention from the emotions surrounding the human values expressed through the act of play. Almost like therapy, players report unparalleled states of concentration and focus during play, making this shift in emotional state one of gaming's biggest takeaways. Whether the activity is fast-action with lots of explosions or slow-paced colored-block matching, players play to experience feelings of excitement, blow away frustration, "get perspective" on their troubles, or create a meditative experience. The stimulation and concentration required drives out bothersome thoughts. Some players simply want to feel more relaxed, excited, or less bored. Players look for a physical and mental workout such as exercise in *Dance Dance Revolution*, heightened reflexes in *Project Gotham Racing*, or an increase in mental acuity from word games such as *Bookworm*. Positive and negative affect from play guides the player toward correct moves, and negative affect increases the perceived pace of the game. Real-world benefits include releasing stress therapeutically, learning new vocabulary or math, improving physical fitness through exercise, and increasing a player's mental agility, which some players believe wards off the effects of aging. Physical movement also creates an emotional release felt after exercise and positive feelings from learning and accomplishing tasks in an engaging way. Games played for any of these reasons offer Serious Fun.

Players play to change themselves. Serious Fun is similar to Easy Fun in that players enjoy immersion rather than challenge. However, one aspect of enjoyment many players prefer is achieving a purpose outside the game experience itself such as the ability of the game to calm or excite them. As in Easy Fun, players want the game to fill their attention, but Serious Fun focuses on players' affects long after the game is over. The term Serious Fun captures the real effort and intent that players expend to alter their internal states, express their values and beliefs, and improve their real-world skills. Without Serious Fun, a game leaves few long-lasting effects and the play experience feels more like a waste of time. Playing may be hard or easy, but what is important to many players is that the effects of a game last long after the game is over.

31.6.2 Chain of Serious Fun Emotions Increases Emotions

In Serious Fun, many players play to change how they feel, and they take different emotional paths to get there (Figure 31.14).

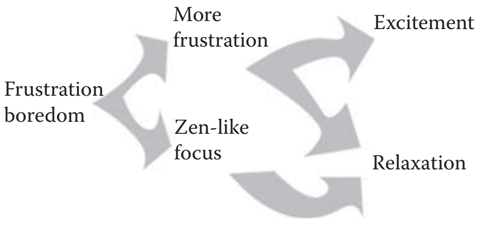
31.6.3 Serious Fun: Enjoyment Helps Accomplish Real Work

"I felt better about playing [crosswords] because it's good for me. If someone would tell me *Tetris* was good for me I'd feel better about playing that."

—Ellen on doing crosswords. She believes the memory demands of the game will keep her mentally sharp and delay the onset of Alzheimer's disease.

Chain of Serious Fun Emotions Increases Enjoyment

Gameplay that changes how a player feels, thinks, or behaves creates Serious Fun emotions from creating something players value.



Emotion	Common Themes and Triggers
Excitement	From sudden changes, novelty, and challenge (Ekman 2003). The unexpected catches attention. Many players enjoy the adrenaline rush. Some want to raise their state of arousal, others like how the emotional intensity makes them more relaxed afterwards.
Relaxation	Relief from negative emotion (Ekman 2003). Gamers often start a game to attain relief from negative emotions and thoughts prior to playing.

* Self-esteem and knowledge acquisition and the result of exercise are not emotions, but changes in a player's internal states that are reported as desirable results from play (Lazzaro 2004b).

FIGURE 31.14 Serious Fun Emotion Cycle. A large part of the appeal of games comes from how they change the player inside. This is referred to as Serious Fun. Here a player starts in one of two emotional states and plays a game to end up feeling different. Many gamers appreciate real-world benefits from play. Some play games to become deeply immersed in a process that they hope will change them long after the game is over. (Courtesy of XEO Design. Copyright © 2005–2007 XEODesign, Inc. All rights reserved.)

Serious Fun teaches or accomplishes real work. Learning-type games, from multiplication to database administration, utilize emotion to encourage the learning of important content or to provide a mental workout, as in an effort to prevent Alzheimer's. Relaxation and excitement make it easier for the player to focus. Serious Fun provides emotions and opportunities for success unavailable to the player in real life by offering a cheaper, safer, more rewarding experience with the content or activity. Rescue simulations can train firefighters about situations that are too dangerous or expensive to do in real time. In educational games such as *DBA Day* (an Oracle workplace simulation XEODesign worked on) players role play a database administrator, including managing their own time. Gee argued that mastering any simulation requires mastering the content embedded in that simulation. By making decisions for the character in a simulation, he believed that simulation games also teach the morals and values of that character coming from their merged identities. The player makes the decisions and the game character has the special abilities to make changes in the virtual world (Gee 2003). These provide learning opportunities, whether it is values and actions of a thief in *Sly Cooper and the Thievius Raccoonus* or a restaurant owner's time and people-management skills in *Diner Dash* (Lazzaro 2005b). Serious Fun provides real benefits from play by using game-like structures to reward concentration and focus attention on an activity that is good for players. Leap Frog's *FLY* pen-top computer helps students with their Spanish vocabulary by offering a verbal translation of words written in English; hearing it spoken in another language motivates practice.

Games offer a new spin on learning by capitalizing on player emotion and interaction to create customized training. *Mavis Beacon Teaches Typing* motivates typists with typing games, such as keeping ants out of a picnic basket or driving a racecar. In addition to explicit learning games (such as Oracle's *DBA Day* or an unrelated project, *Doom DBA*), games also embed techniques to offer players more immersive instruction. The introductory experience in *Halo* not only provides a seamless introduction to the user interface by integrating it into the story, but also adjusts the controls to meet the player's preferences without a dialog box. In *Grand Tourismo*, a detailed car-racing simulation game, the opening time trials break down racing skills, such as cornering, into small, easy-to-master steps that fit together to create a more enjoyable practicing experience. By providing a motivating alternative to accomplishing an otherwise boring or unmotivating task, Serious Fun helps players accomplish real-world objectives such as getting in shape. Someone may play *Dance Dance Revolution* for the excitement of moving to the music, to lose weight by burning calories, or to learn the physical skills and coordination required to dance better. Others play *Karaoke Revolution* to learn how to sing.

Games even make work fun. Players pay for the experience of being a waitress (*Diner Dash)*, business owner (*Lemonade Tycoon*), dungeon master (*Dungeon Keeper*), professional sports player (*Madden Football*), or theme-park owner (*Rollercoaster Tycoon*). They even buy the opportunity to sort bugs by color (*Tumblebugs*), pick up their rooms (*Katamari Damacy*), or manage a city (*Sim City*). Games model life problems (*The Sims*) and improve performance during real work. In *The ESP Game*, developed at Carnegie Mellon University, people play a guessing game to enliven the otherwise boring task of providing text labels for images (von Ahn and Dabbish 2004). Idea or prediction markets use games to beat expert opinion polls. Prediction markets allow participants to express their opinions through buying and selling shares with either virtual or real money. One of the oldest, the University of Iowa's *Iowa Electronic Market*, has allegedly beat expert polls in predicting U.S. presidential elections since 1988. Players of the *Hollywood Stock Exchange* use virtual money to predict what actor, director, or film will receive an Oscar nomination. At *NewsFutures*, players compete for prizes based on their ability to predict news events. Several companies, such as Google, use internal markets to predict launch dates and job openings. Other companies such as Newsfutures/Lumenogic have created public prediction markets and led the development of enterprise-class prediction market services. In a landmark study in 2004, it was demonstrated that play-money markets can be just as predictive as real-money ones (Servan-Schreiber et al. 2004). Not without controversy (gambling is illegal in many countries), markets have even been proposed to predict terrorist attacks (Hulse 2003). Because playing prediction markets accomplishes a real purpose, it changes how participants feel about playing in general. Like offering a prize for a competition to develop a solar car, having real money or reputation on the line increases the excitement. In Serious Fun, players accomplish real work for many reasons, including to meditate, lose weight, label every image on Internet, or beat Wall Street predictions.

31.7 AMUSEMENT FROM PEOPLE FUN GAMEPLAY

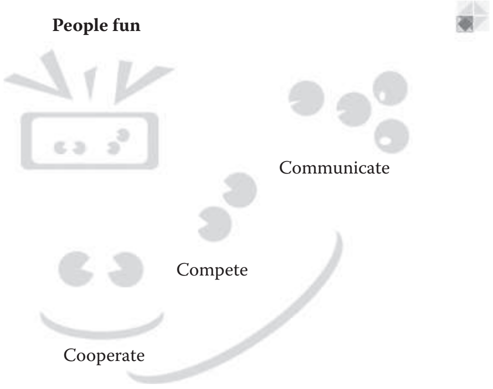
31.7.1 Amusement Encourages Social Interaction

> "It's the people that are addictive, not the game."
>
> **—Bob, a hard-core sports game player**

One advantage that a computer game has is that it is ready to play when you are. No friend required. Still, for many people it is the experience of playing with friends or family that makes play worthwhile. Group gaming provides a mechanism for social interaction, a quick excuse to invite friends over, and more challenges to gameplay. People Fun creates amusement between gamers as they play to spend time with their friends. People report playing games when their friends do, playing games they do not like, and even playing the types of games that they do not like playing just to spend more time with their friends (Lazzaro 2004b).

> "Since we lost half our guild to Star Wars Galaxies it's not as fun."
>
> **—A hard-core gamer playing *Dark Age of Camelot***

People Fun is a self-motivating activity that maintains player engagement by supporting interaction with other people (Figure 31.15). People Fun encourages interaction with other players as they establish social hierarchies, joke, and develop social bonds. It creates affect by providing opportunities to cooperate, compete, and watch others play. Some emotions, such as gratitude, generosity, and *schadenfreude*, the German word for "pleasure at the misfortune of a rival" (Ekman 2003), cannot occur while playing alone. When people play together they invent new ways to interact, develop house rules, and add their own content to create more pleasure for themselves and their friends.



Game Features
cooperate
compete
communicate
perform
spectacle
characters
personalize

Feelings
amusement
shadenfreude
naches
generosity
gratitude
elevation

Gameplay that involves others to create emotion between players and increase social bonds.

FIGURE 31.15 People Fun creates emotions that increase a player's enjoyment from social interaction. The player experience profile of People Fun includes many emotions that require two people such as schadenfreude, amusement, naches, and gratitude. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

The most emotion from games arises when people play together in the same room. Whether playing in a living room or cyber café, the frequency, duration, variety, and intensity of emotional displays all increase (Lazzaro 2004b). The emotions of schadenfreude, amusement, and *naches* (Yiddish for the pleasure and pride felt when a child or mentee succeeds) occur more frequently when playing with others. Playing over the Internet, even with video or voice chat, elicits fewer emotional displays than when playing in the same room. A major reason behind this increased emotion is that being in the same room allows for additional interaction between players beyond what is available in the game alone. Players exchange insults and outdo each other with witty commentary. They add content and new rules as real time presence allows for more flexibility (Lazzaro 2004b). Considered in light of Ekman's nine pathways to creating emotions (Figure 31.16), People Fun uses more of them than Hard Fun, Easy Fun, or Serious Fun. In fact, the last five out of the nine in Figure 31.16 are enhanced in group face-to-face play.

31.7.2 People Fun: Creates More Emotion When Playing Face to Face

> "Enjoyment when your friends blow up."
>
> **—Pat, a PS2 gamer on his favorite game emotion**

The involvement of others increases the complexity of the game by creating cooperation and competition, which in turn increases the emotions from play. Different roles for players with shared and opposing objectives increase player interaction. Achieving a goal with a simple rule set becomes exponentially more complex when a player's opponent is another person or group. Not all players like the feelings of rivalry from competition with their friends. Many prefer to team up with their buddies to compete against computer-controlled characters. To encourage group formation, Massively Multiplayer Online Games (MMOs) often have interdependent classes of players, a design that comes from the *Dungeons and Dragons* paper-based role-playing game. In these games, such as *WOW*, character classes encourage the formation of groups, and treasure quests and dangerous dungeons require players to pool their characters' specializations such as archers, swordsmen, hunters, paladins, wizards, and healers. Improved success rates and score bonuses reward players working together in the game and help form the social bonds that increase a player's emotional attachment to the game (Figure 31.17).

Nine Pathways to Emotions

(Ekman 2003)	Example from Games
1. Auto appraisal	Explosions, fire, and boiling lava
2. Reflective appraisal	Thinking about last night's dungeon raid
3. Memory of an emotion	Remembering falling from a cliff
4. Imagination	Thinking about what happens from falling off a cliff
5. Talking about	Discussing game events with other players
6. Empathy or witnessing another's emotion	Another player's facial expression or character emote
7. Instruction by others on how to feel	Another player's assessment of an event
8. Violation of social norm	Driving over other players instead of racing against them
9. Making a facial expression of an emotion	Smiling and laughing after defeat in front of friends makes it feel more positive

FIGURE 31.16 There are nine ways to create emotions in a person.

People Fun keeps players going by providing new ways for players to interact. It provides a variety of player-to-player outcomes and bright spectacles for the audience. It provides the opportunity to perform difficult maneuvers and stunts that reward practice with which to amuse and amaze friends. In *Soul Calibur II*, players interact with each other by fighting rather than racing side by side to beat the clock. The frequency and the variety of player-to-player interaction increase commentary between players and make it more enjoyable for an audience to watch. Spectators can participate throughout with comments and criticisms. People Fun games with hidden aspects, special moves, cheat codes, and Easter Eggs (hidden games, objects, rooms, or animations) offer even more opportunity to impress friends. All of these create admiration and envy between players (Lazzaro 2004b).

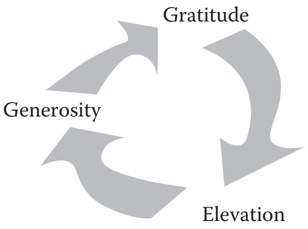
31.7.2.1 Some Emotions Require Two People

People Fun wins out over other types of gameplay with its exclusive access to emotions that require two people, such as schadenfreude and naches. The most frequent emotion in co-located group gaming is amusement. In a group context, even negative events are sources of laughter. The feeling of naches and emotional attachment created in *Nintendogs* (a puppy simulation game for the Nintendo DS) are so strong that they not only created a best-selling title, but the title's popularity increased the sales of the hardware platform as well.

People Fun in games is also present in interaction with characters on screen. It enhances emotions with facial expressions

Chain of People Fun Emotions Increases Enjoyment between Players

People Fun motivates group interaction, interpersonal relationships, and creates emotions between players.



Emotion	Common Themes and Triggers
Amusement	Something that's funny. (Ekman 2003). Regardless of the emotional tenor of the game, people in groups laugh more than when playing alone.
Schadenfreude*	Pleasure over misfortune of a rival (Ekman 2003). Competitive players enjoy beating each other, especially a long-term rival. Boasts are made about player prowess and ranking.
Naches*	Pleasure or pride at the accomplishment of a child or mentee (Ekman 2003). Players enjoy seeing friends succeed as well as the achievements of nonplayer characters they have helped, such as in *Back to Baghdad*, where players can exchange health packs.

* Schadenfreude and naches do not have universal facial gestures; however, they are frequently described by players (Lazzaro 2004b).

FIGURE 31.17 People Fun Emotion Cycle. In People Fun, some emotions link together like a chain between players, such as offering a healing spell or health pack (generosity) to a fellow player in need (gratitude). A third player feels elevation at witnessing this human kindness and becomes more likely to be generous. Next time a different player may be in need and the links on this chain would rotate. In this way, one game mechanic offers three emotions depending on who starts the chain. (Courtesy of XEO Design. Copyright © 2005–2007 XEODesign, Inc. All rights reserved.)

for nonplayer characters (NPCs) as well as for player-controlled characters. In *Diner Dash*, players not only have to react to changes in characters' facial expressions but to win the game requires managing customer emotions to prevent them from getting so angry that they leave (Lazzaro 2005b). Games such as *World of Warcraft* increase emotions between players by offering emotes and chat features. Adding voice and video communication between players increases emotions, and co-located group play takes this one step further by supporting face-to-face collaboration in addition to face-to-screen.

31.7.3 Players Modify Games to Enhance Emotions during Group Play

In People Fun, players like playing together so much that they create house rules and convert single player games to make group play more enjoyable. In order to play a *Buzz Lightyear* game together, one family assigns each person a different key to run, jump, or shoot. To share in a pirate adventure, one set of three college students plays *Sid Myer's Pirates!* game by taking turns. In a bike-racing game where players outnumbered the controllers, the winner is the player who "ran the gauntlet" and beat all challengers in the room, not by winning the race, as that would take too long, but by out-distancing each opponent. This said, some players eschew contact with other people if they mostly care about developing strategy or understanding how to play better. To them, playing a game such as *Hearts* online is better when played alone because they do not want to chat or posture with players that they do not know. Such players prefer the Hard Fun even in multiplayer games. Whether it is *WOW* or *Spades*, and depending on their mood, some end up soloing rather than joining a group.

31.8 EMOTION FROM ENTERTAINMENT INTERFACES AND INTERACTIVE PRODUCTS

The Four Fun Keys—Hard Fun, Easy Fun, Serious Fun, and People Fun—play an important role in making games self-motivating. They provide the opportunity for challenge and mastery, prospect for exploration and discovery, give a boost to self-esteem and an excuse to spend time with friends. The gameplay in games capitalizes on the interaction between human emotion, thought, and behavior. From this perspective, productivity applications have the same goal. Productivity applications take for granted that the user provides the motivation for using the tool, such as workplace responsibilities or a boss's deadline. The compelling and enjoyable nature of computer games makes the process itself enjoyable, so game designers spend a great deal of design effort to ensure that their games provide a rich emotional experience during play. Study of these interactions offers clues as to how to make productivity tools more motivating without making it a game. The importance of emotions in design is that emotions can influence when, how, and why to use a product.

31.8.1 Game-Inspired Affect Makes Product Use More Entertaining

Games offer many lessons for product-interaction design outside of entertainment, especially in the relationship between design and emotion. Software should make tasks easy by requiring just a few steps and providing appropriate emotional stimuli to focus and reward the user for completing them. One of the biggest differences between productivity software and software used for entertainment is that the pleasure and motivation come from accomplishing outside tasks, such as completing a spreadsheet or creating a database. Emotions in nongames should create enjoyment, aid decision making, focus effort and attention, and provide motivation. Like games, it is possible to design interactions that are similar to all four types of fun to create a wide spectrum of emotions in nongame software (Figure 31.18).

31.8.1.1 Playful Product Attributes Increase Emotions during Use

Many products employ novel opening techniques to increase customer appeal. Beyond "out-of-the-box" experiences, the experience of the product in use is becoming an important part of a product's competitive advantage. Similar to the way packaging frames consumer expectations and emotional associations with a product, how a product opens or switches on can create emotions and other associations every time the product is used. For example, Danger's *Sidekick* cell-phone lid spins open horizontally with a snap. The *Robo-Book* Light surprises by unfolding like a mechanical arm. The nozzle on *All* liquid laundry soap dispenser features a bright red rubber

Creating Affect through Actions for Nongames

Hard Fun

Affect to increase focus on and enjoyment of a task:
Spreadsheets
Word processing
Time management
Decision support

Easy Fun

Affect to capture attention and enjoyment:
Data mining
Searching
Multivariant problem solving
Creativity support

People Fun

Affect to facilitate the interaction and cohesion between individuals:
Cell phones
E-mail
Human-resource and project-management applications
Groupware
Presentation software

Serious Fun

Affect to help user to accomplish a lot of work consistently over time:
Learning tools
Therapies
Medical devices
Cleaning products

FIGURE 31.18 The affect from the Four Fun Keys can increase emotions and enjoyment for several kinds of productivity-software applications and consumer products. (Courtesy of XEO Design. Copyright © 2004–2007 XEODesign, Inc. All rights reserved.)

ball to squeeze for detergent. The designers at IDEO transformed the tiered old carpet sweeper to create the *Swiffer Carpet Flick*. The *Carpet Flick* combines a free-flowing universal joint at the head (Easy Fun), a transparent body so customers miss none of the action (Hard Fun's feedback on goals), and accomplishes real work in an enjoyable way (Serious Fun). The buttons on some productivity software offer Easy Fun if they glow or highlight when moused over. A hard disk defragmenting utility displays progress with an animation that is mesmerizing and game-like to watch, as the colored segments are organized and grouped together. None of these products are games, yet all of them offer Easy Fun with novelty to inspire curiosity, surprise, and a little wonder. Regardless of whether it is a productivity application or a trash can, if it creates enjoyable emotions, it will be used more often and shown off to friends (People Fun).

Breath-relaxation techniques are more effective if practiced daily. To encourage frequent use with Serious Fun the *Stresseraser* (a home-therapy device that provides bio feedback on the user's pulse and breathing exercises) uses both Hard Fun and Serious Fun. The device rewards customers with a point system and game-like chart feedback of their pulse data. The use of a display graph and a gradual increase in number of points earned per session encourages use by offering Hard Fun and Serious Fun at the same time.

> "Why not challenge yourself to see how good you can make yourself feel by getting 100 or more points a day for two months?"
>
> "A session of 30 or more points is a great way to start the day off right. Especially on days when you feel like you got up on the wrong side of the bed."
>
> **—www.stresseraser.com**

The catalog copy on the company's website frames customer expectations in terms of Hard Fun. The game-like intent of the product designers is even more evident in unreleased game modules where customers float a bird around obstacles as a reward for reducing their pulse (Fabricant 2005). Their approach to relaxation is not about building frustration to eventually succeed through their skill to feel fiero. Instead, with the *Stresseraser*, as in the Interaction Institute's *Brain Ball* game (Hjelm 2003), people win by relaxing and taking deep breaths. For these products goals and feedback allow players to monitor and change their own emotional state for Hard Fun and Serious Fun.

31.8.2 Designing Game Affect for Entertainment Access

31.8.2.1 Playful Interaction and Feedback Facilitates Entertainment Access

Game-inspired affect increases the appeal of entertainment applications such as televisions, game consoles, e-zines, and music players. Because the primary customer goal is to locate content, accessing entertainment options need not be a game in itself. However, entertainment devices can have a game-like feel. They should be even easier to use than productivity applications, because they don't borrow motivation from completing an outside task. They should also create excitement about the entertainment options they provide.

Since the beginning, graphic-user interfaces have employed metaphors such as windows, trashcans, and elevators to explain how features work. Apple's Macintosh OSX operating system established a new trend in modern interface design by going beyond the clear explanation of function to creating an experience. Mac OSX is an interface that is "fun" to use. The dock magnifies icons under the cursor to make them easier to select, at the same time creating pleasurable animation that is pure Easy Fun, even without the goal of opening an application. The genie-out-of-a-bottle animation of document windows as they expand out of a tiny icon on the dock offers more Easy Fun to create curiosity, surprise, and wonder. Operating these features entertains all by itself.

31.8.2.2 Creating Emotions in Entertainment Access: Apple's iPod Case Study

The most game-inspired device to come out of Apple is the iPod. The iPod's novel combination of hardware and software interfaces creates a new experience full of affect. The novel interaction facilitates entertainment access by offering a customer experience unlike any other. While not technically a game, the iPod uses Hard Fun, Easy Fun, Serious Fun, and People Fun to create emotional experiences for the user.

Central to the user experience of the iPod is the control wheel. It attracts attention and creates emotions of Easy Fun such as curiosity and surprise. Its novel round motion easily scrolls through long play lists. Emotions arising from exploring the free-flowing motion focus attention on finding a song. The quick feedback (including a separate speaker just for the scroll clicks) enhances the sense of mastery (Hard Fun) as a customer gains control. It feels like a game. Even without pressing play, scrolling is enjoyable because using the four directional buttons and wheel mimic a game controller. In addition to the form resembling a stereo speaker, the circular DJ Scratching action takes advantage of an action already associated with making music more exciting and even offers an opportunity to role play for more Easy Fun. The music visualizations on the companion iTunes software offer fascinating animations in time with the music.

A game of *Bricks* (similar to *Breakout*) shipped as an Easter egg (hidden game) in the first edition and later more elaborate games such as Zuma under their own games tab further established the iPod's connection to fun and enjoyment. As easy as the iPod is to use, users must experiment and might experience frustration in learning to understand the menu hierarchy. This offers opportunities for Hard Fun as customers search for their favorite track. Users may experience a feeling of accomplishment rather than fiero when they succeed, but the dancing black silhouette models in the iPod's ad campaigns frolicing with arms overhead clearly evokes fiero and other kinds of joy. That the iPod offers Serious Fun of mood-altering music and videos is a given. For additional Serious Fun, iPods also play business-audio books and offer file storage to do the real work of transporting documents and spreadsheets along with music files. The trademark white headphones offer emotions from People Fun even while separating from others auditorally by sending a clear message about the fun-loving social group to which an iPod customer belongs. These status-symbol icons create People Fun emotions from association with belonging to the "in crowd" as well as the emotions expressed in the ads. Several iPod accessories allow customers to share their music during playback. And iTunes' trading of play lists, creation of custom CDs, and podcasting all provide ways for iPod users to reach out to other people for more People Fun.

31.8.2.3 Creating Emotions in Entertainment Access: Microsoft's Xbox 360 Case Study

31.8.2.3.1 Ready to Inhale

The designers of the Xbox 360 video game platform had a specific customer experience in mind. Whereas the design inspiration for the first Xbox was like the Hulk bulging with muscles to break out of a box, the Xbox 360 was to be more of an inhale, like Bruce Lee drawing breath in preparation for play, poised, and ready to strike. Instead of a file-and-folder metaphor, the main UI for the video-game console is designed to capture a moment of preparation before the challenge. The idea was to avoid fussy animations, yet provide something interesting and intuitive to get the player emotionally ready and mentally focused before play begins. The Easy Fun of navigating the menus increases immersion. The emotions for this interface match the player's goals during this part of the use cycle.

In terms of Easy Fun, the menus for the Xbox 360's main UI curve sideways like rib bones and mimic the gesture of inhalation as they move. The menus slip from side to side in a breathing-like motion. The sounds are organic and whip-like with a satisfying thump at the end as the menu snaps into place. This novel auditory and visual experience makes the menus enjoyable to explore without becoming too complicated or intrusive. Novelty of the sideways menus creates curiosity and encourages exploration. The designers were going for playful (the attitude in games) without being so entertaining as to be a game in itself. It is about providing very simple access and a pleasurable experience that creates excitement but doesn't get in the way.

The most important way the Xbox 360 increases enjoyment of games played on its console is through the use of People Fun. To deliver more People Fun emotions and to foster connections between players, the Xbox 360 offers community features for all its titles. Gamers can create their own persistent player profile (or gamer card) that spans game titles. Personal profiles provide an in-game identity and list in-game accomplishments so friends can compare game-specific achievements, scores, and what game they are currently playing. For the nongamers in the household, the Xbox 360 allows access and display of family photos and video for a different kind of social interaction.

The emotions in these user experiences for iPods and Xbox 360s not only help customers accomplish tasks. The UIs create experiences that make the devices pleasurable to use, put users in the appropriate emotional and cognitive mindset, and capture the next level of the four pleasures that Jordan considers essential in the new human factors: (a) Physio-pleasure, (b) Psycho-pleasure, (c) Ideo-pleasure, and (d) Socio-pleasure (Jordan 2000). As the focus for HCI shifts from interface design to user experience, more accurate taxonomies for internal experiences and methods to measure these experiences are needed. The development of these tools will facilitate the design of more enjoyable player experiences. While entertainment interfaces such as the iPod and the Xbox 360 have done much to create powerful user experiences on several levels, there is still much more that is possible to do.

31.9 NEW DIRECTIONS AND OPEN ISSUES

31.9.1 New Play with Smaller, Smarter, More Flexible Devices

As technology shrinks and integrates itself into more aspects of our lives, people play more games in more places. Devices that are smaller, smarter, mobile, and contextually aware create new opportunities for electronic gaming, which until now has largely been restricted to players at home around a single screen. What happens when the services offered by a laptop become cheap enough to print on a candy wrapper? Mobile and alternate-reality gaming offer the opportunity to make things happen in the real world, which further enhances other experiences such as enjoying the company of friends and dispelling boredom while waiting in line. Games break out into the real world through geo-caching and games of tag through a city. Promotional Alternate Reality Games (ARGs), such as the *The Beast* for the movie *AI* and *I Love Bees* for the game *Halo 2*, use real-world-web-enhanced gameplay to market products and create communities of millions of players (4ourty2wo 2005). With ubiquitous computing, everything from mobile phones, to the front door, to a ketchup bottle in a diner could contain enough smarts to offer services. Will they all contain a screen and therefore the potential to host a game? Will we surf the net from our saltshaker or will it provide other opportunities to engage our attention?

Many designers chase faster hardware and better graphics, yet the emotional power of games does not occur on screen; it takes place inside the player's head. The biggest emotions that new technology will create is not through rendering blood, sweat, and tears in molecular detail. The real changes in emotional gaming will happen through supporting a more agile design process and by making gaming devices smaller, sharable, context-aware, and more accessible, such as game controllers with fewer than 12 buttons like Nintendo's *Wii*. Until we all get personal holodecks and natural language processing, the success of many new kinds of games will come from connecting the cheapest emotion generators around: a player's rivals and friends.

Life is becoming more game-like as games and elements of play inspire new dimensions of product appeal. Like games, products create experiences. Cutting-edge product designers now design customer experiences, not products, as witnessed by the iPod. Designers aim for engagement in addition to making something better/faster/cheaper and easier to use or market. Adding playful elements to goods and services increases the attraction of everything from advertising messages to Southwest Airline's in-flight safety announcements to the design of public spaces. We see the increasing importance of emotions in design of products such as the playful squid shape of the Phillip Stark orange juicer, the surprising opening rotation of Danger's *Sidekick* mobile phone, and in the pleasing octave chords produced by *Segway's* acoustically designed motors. The progress being made toward this shift toward more emotional design is even more apparent in games.

Games are already redesigning how we work and shop. Employers use games to screen potential hires for three-dimensional (3D) reasoning skills and train them to solve problems with multiple variables. Games changed consumer processes such as buying and selling on eBay or the dining experience at Dave and Busters (Chuck E. Cheese for adults). Even web software applications like Flikr (www.flikr.com) include more fun to increase appeal by sharing items between friends or by offering features to be "gamed," such as displaying the number of friends a person has in *Friendster* or *LinkedIn* (www.friendster.com; www.linkedin.com) social-networking software. Understanding how games create emotions makes products and services more engaging and enjoyable and even improves community and the quality of life.

31.9.2 Open Issues

Better design of the emotions required for and that result from interaction will unlock more human potential from using products and games by priming and rewarding users with appropriate affective states. We know that cognition and emotion walk hand in hand. We are only just beginning to discover how they support each other and how interaction creates emotion. The Four Fun Keys model starts this journey to build and examine player experiences in terms of specific emotions such as fiero from Hard Fun, curiosity from Easy Fun, relaxation from Serious Fun, and amusement from People Fun. Further measurement of specific emotions during different types of cognitive, behavioral, and emotional experiences will help piece together the role individual emotions play in cognition and activities. For example, a surprising event orients an individual's attention to determine whether the source is a benefit or a threat. Surprise does this in addition to creating internal sensations, producing facial expressions, and communicating information about the source to others. Curiosity, the lead emotion from Easy Fun, has a strong cognitive component, which also focuses attention in a pleasurable way. Satisfying curiosity especially when it results in feelings of surprise and wonder produces strong pleasurable sensations that motivate an individual to repeat the activity. Additional research will tell us how to create designs that inspire, maintain, and intensify feelings of curiosity. This is important for applications such as *Google* to improve searching the Internet or when browsing an e-commerce catalog (www.google.com). Conversely, intensifying fiero from finding the search item also improves the user experience. Informed with a deep understanding of how curiosity and frustration employ different affective states to focus attention, the design of entertainment and productivity products can achieve the next level of engagement by sculpting emotional responses that complement the tasks.

31.10 CONCLUSION

Games have the creative freedom to push technical boundaries and be light years ahead of productivity interfaces. They have been champions and early adopters of new interface techniques from joysticks to voice commands. Game interfaces lasso new hardware and experiment with dialogs and menus to deliver novel experiences with a vigor never seen in productivity applications. Mastering these innovative interaction techniques provides richer experiences that inspire a devotion to learning features. Radial menus in *The Sims*, audio menus, and draw-your-own-game interfaces in LeapFrog's *FLY* pen-top computer, the Line Rider web game, the iPod's scroll wheel, and Xbox 360's organic side-scrolling tabs are all examples of original alternatives to navigating file-and-folder hierarchies and dialog boxes. Games boast heads-up displays and voice commands to allow players to multitask (*Tom Clancy's Rainbow Six* on the Xbox) or sing (*Karaoke Revolution*). Still other games use camera interfaces that track the body's movement (*EyeToy* games, *Virus Attack* for camera phones), physical motion (*Dance Dance Revolution* dance pad, Wii Sports), and touch and gestures (*Black and White, Nintendogs, Yoshi's Touch and Go, Electroplankton, Pac-Pix*). Others use positional sensing/ubiquitous computing and motion sensors (*WarioWare: Twisted*). Games offer rich social systems for interpersonal collaboration and communication (*WOW, EverQuest*). Games even prototype future interface technology such as context-sensitive holograms to encourage cursor exploration (*Star Wars Commando*). The interface for each of these technologies supports a fresh experience.

In addition, games offer an emotional punch that has become a cultural force. The automotive industry already consults racing-game designers on how to make more exciting cars. U.S. politicians such as Howard Dean use web games to teach democracy and increase interest in their campaigns. Games are not only innovating interface design, they are also full of emotional experiments. For example, in *Fable*, the moral values expressed by player decisions change the character into a shining, blonde-haired hero or demonic devil with horns. Although not everyone plays computer games, they command a profound influence on other media and culture. Additionally, the generation raised on games (today's college students all played *Oregon Trail* in grade school) will play more games as adults than their parents did.

Games' distinct emotional processes come from what players like most during play, which informs the functional design of product and software. Unfortunately, the sole emotion goal addressed by usability is to reduce frustration. While important, this falls short of offering strong emotional rewards for accomplishing difficult tasks. Crafting these emotions contributes to user success because, when inspired by rich emotional responses, users will explore and learn more of an application's features, making them more efficient at their jobs. Over-engineering a task by making it too predictable, repetitive, and easy to complete increases the likelihood of boredom and actually reduces satisfaction over the long term. Players and workers experience huge emotional rewards for completing complex challenging tasks. It would be unfortunate if these were eliminated from the work we do as humans.

In HCI, the big mistake used to be fixing clumsy complex features with pretty background graphics. Today, the big mistake lies in believing that user experiences will improve if designers remove frustration points (usability), have the interface do all the work, and reduce the task to a series of trivial steps. Customers already have emotional reactions to their software. Designers must learn to speak this language of emotion from interaction.

The role of a good interface is to prepare users to master something difficult, then allow them to give themselves credit for mastering the difficult skill, while leaving enough ambiguity and challenge to make the task fun. What's next for user-experience design goes beyond refining features to logically support how users perform tasks. Design must also address how features motivate users through affective states to support and refresh the task at hand, not by making the software unnecessarily complicated, but by doing what games do: support sequential skill-building to achieve complex goals and encourage players to move beyond their points of failure to feel empowered masters of their own destiny. Experienced designers are already using Hard Fun, Easy Fun, Serious Fun, and People Fun by embedding emotions from goal-directed, open-ended, purposeful, and social play.

By offering life problems in miniature, games provide important clues to the relationship of emotion in human problem solving, goal achievement, and interaction with other people. Researching games clearly defines the relationships between action and emotion as well as between emotions themselves. In games, many emotions are opposites, happen in sequences, have prerequisites, and share links with others. Some emotions require people, relate to goals, or involve the future or the past. Connecting how gameplay leads to specific emotions establishes a framework for the process of creating more emotional user experiences. The Four Fun Keys Model creates effective analysis and design techniques to identify and create a wider range of emotions. It connects emotions to popular types of gameplay and demonstrates how chains of emotions are embedded or released from different activities. By harnessing play, even productivity software and workflow design can take advantage of the motivating force of game emotions.

Games are the new medium of the twenty-first century. Unlike any other design discipline, player experiences that unfold over time are more interactive than movies, painting, architecture, industrial design, literature, or fashion. Games are dynamic processes that create dynamic experiences. They are much more than a series of static impressions seen in sequence. Games offer a unique set of emotions from accomplishment and failure. At their core is the ability for players to interact with the content. The experiences that come from this interaction, the ability to create emotions, and the way that this interaction moves will separate games as an art form distinct from movies and other visual arts.

Emotions play an essential role in providing the entertainment value that captivates players. From games, we can learn how to improve the emotions that keep users engaged in activities of different types. Through player interaction and control of events, games promise to become more emotional than movies and other entertainment, but whether these are the same emotions remains to be seen. Games now include more detailed storylines to increase player engagement. However, it is clear from XEODesign's research that games create strong emotions even without stories, and it is clear that games already create more emotions through interaction with other players in the game world. Many aspects of the storytelling language of cinema apply to games and they will still move players emotionally. However, emotions from the player's goals and the things he or she cares about will likely prove to be stronger. In games, the mechanic is the moral of the story.

Releasing the full emotional potential of games will not be easy. It requires deep understanding of how emotions work, because more is known about crafting emotional entertainment experiences (such as a movie) by offering the viewer empathy with a protagonist engaged in a predetermined sequence of events. Less is known about creating emotion through being the protagonist. Viewing a prerendered video between game levels is less compelling than having the player's actions create an emotionally rich experience itself, but at the moment these mini-movies are easier to design. Going forward, these dramatic tools for creating affect should inspire rather than dictate game design. Eventually the emotional props from these cut scenes will fade from games just as title cards disappeared from old silent pictures once sound technology allowed the actors and the action to speak for themselves.

Psychology, sociology, theater, literature, film, and only recently games have all studied how entertainment engages audiences. Games are unique in that they entertain by creating emotion from interaction. Before XEODesign's research on emotion and the fun of games, none had studied how entertainment creates specific emotions and uses them to focus attention and motivate play. The Four Fun Keys Model creates the four most important sets of emotions for games. Each fun key is a collection of game interactions (game mechanics) that captivate player attention with different series of emotion that make games self-motivating. These emotion cycles create the player experience and separate best-selling games from their imitators. Based on contextual research of people playing their favorite games cross genre, platform, and gender, the Four Fun Keys describe how emotion comes from what is the most fun about games. Entertainment interfaces and products such as Apple's iPod and Microsoft's Xbox 360 already use interaction from each of the Four Fun Keys to unlock powerful emotions to increase user enjoyment and build stronger brand impressions. Interaction that generates a lot of emotion is more memorable and feels like play.

ACKNOWLEDGMENTS

Jane Booth, Brian "Psychochild" Green, Ron Meiners, Steve Meretzky, and Jeff Pobst provided detailed feedback on drafts of this chapter. Hal Barwood, Lura Dolas, Noah Falstein, Bill Fulton, Lee Gilmore, Russ Glaser, Kevin Keeker, Paolo Malabuyo, Marcos Nunes-Ueno, Randy Pagulayan, Patricia Pizer, Kent Quirk, Brian Robbins, Ann Smulka, Mark Terrano, Gordon Walton, and Richard A. Watson all provided useful guidance, suggestions, and examples.

REFERENCES

4ourty2wo. 2005. The beast. http://www.42entertainment.com/beast.html (accessed November 29, 2011).

Bartle, R. 2003a. *A Self of Sense*. http://www.mud.co.uk/richard/selfware.htm (accessed December 29, 2005).

Bartle, R. 2003b. *Designing Virtual Worlds*. New Riders Games. Berkeley, CA: Peach Pit Press.

Boorstin, J. 1990. *Making Movies Work*. Beverly Hills, CA: Silman-James Press.

Brave, S., and C. Nass. 2002. A. Emotion in human–computer interaction. In *The Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, ed. J. Jacto and A. Sears, 81–96. Mahwah, NJ: Lawrence Erlbaum Associates.

Clore, G. C., and K. Gasper. 2000. Feeling is believing: Some affective influences on belief. In *Emotions and Beliefs: How Feelings Influence Thoughts*, ed. N. H. Frijda, A. S. R. Manstead, and S. Bem, 10–44. Paris/Cambridge: Editions de la Masion des Sciences de l'Homme and Cambridge University Press (jointly published).

Csikszentmihalyi, M. 1990. *Flow: The Psychology of Optimal Experience*. New York: Harper & Row.

Damasio, A. 1994. *Descartes' Error: Emotion, Reason, and The Human Brain*. New York: Quill Penguin Putnam.

Ekman, P. 2003. *Emotions Revealed*. New York: Times Books Henry Hold and Company, LLC.

Fabricant, R. 2005. HRV monitor: Creating a guided user experience on handheld devices. In *Proceedings of DUX 2005*. San Francisco, USA.

Gee, J. 2003. *What Video Games Have to Teach us About Learning and Literacy*. New York: Palgrave Macmillan.

Hassenzahl, M., A. Burmester, M., and K. Lehner 2000. Hedonic and ergonomic quality aspects determine a software's appeal. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interation Conference (CHI)*, 201–8. The Hague, the Netherlands. New York: ACM.

Hazlett, H. 2003. Measurement of user frustration: A biologic approach. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction Conference (CHI)*, 734–5, April, 2003.

Hjelm, I. 2003. BrainBall research + design: The making of Brainball. *Interactions* 10(1):26–34.

Hulse, C. 2003. Threats and responses: Plans and criticisms; Pentagon prepares a futures market on terror attacks. The New York Times. Retrieved November 29, 2011, from http://www.nytimes.com/2003/07/29/us/threats-responses-plans-criticisms-pentagon-prepares-futures-market-terror.htmlpagewanted=all&src=pm.

Jordan, P. W. 2000. *Designing Pleasurable Products: An Introduction to the New Human Factors*. London: Taylor & Francis.

Kim, A. J. 2000. *Community Building on the Web*. Berkeley, CA: Peach Pit Press.

Lazzaro, N. 2004a. Why we play games. *User Exp Mag* 8.

Lazzaro, N. 2004b. Why we play games: Four keys to more emotion in player experiences. In *Proceedings of the Game Developers Conference*. San Jose, California, USA. www.xeodesign.com/whyweplaygames.html (accessed December 28, 2005).

Lazzaro, N. 2005a. *Design survey of game designers*. Unpublished manuscript.

Lazzaro, N. 2005b. *Diner dash and the people factor* from www.xeodesign.com/whyweplaygames.html (accessed March 2, 2005).

Lazzaro, N., and K. Keeker. 2004. What's My Method? A game show on games. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction Conference (CHI)*, 1093–4. Vienna, Austria.

LeBlanc, M., R. Hunicke, and R. Zubek. 2004. MDA: A formal approach to game design and game research. http://www.cs.northwestern.edu/~hunicke/pubs/MDA.pdf (accessed March 2, 2005).

Malone, T. 1981. Heuristics for designing enjoyable user interfaces: Lessons from computer games. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer Human Conference (CHI)* 63–8.

Mandryk, R. 2004. Objectively evaluating entertainment technology. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction Conference (CHI)*, 1057–8. Doctoral Consortium, Vienna, Austria.

Norman, D. A. 2004. *Emotional Design: Why We Love (or hate) EveryDay Things*. New York: Basic Books.

Papert, S. 2005. *Hard Fun*. http://www.papert.org/articles/HardFun.html (accessed December 29, 2005).

Piaget, J. 1962. *Play, Dreams, and Imitation in Childhood*. New York: Norton.

Servan-Schreiber, E., J. Wolfers, D. M. Pennock, and B. Galebach. 2004. Prediction markets: Does money matter? *Electronic Markets* 14(3):243–51. Retrieved November 29, 2011, from http://bpp .wharton.upenn.edu/jwolfers/Papers/DoesMoneyMatter.pdf

Sykes, J., and S. Brown. 2003. Affective gaming: Measuring emotion through the gamepad. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction Conference (CHI),* 732–3. New York: ACM.

Tiger, L. 1992. *The Pursuit of Pleasure*, 52–60. Boston, MA: Little, Brown & Company.

Thorson, E., and M. Friestad. 1985. The effects of emotion on episode memory for television commercials. In *Advances in Consumer Psychology* , ed. P. Cafferata and A. Tybor, 131–6. Lexington, MA: Lexington.

von Ahn, L., and L. Dabbish 2004. Labeling images with a computer game. In *Proceedings Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction. Conference (CHI)*, 319–26. Vienna, Austria, New York: ACM Press.

Wright, P., J. McCarthy, and L. Meekison. 2003. Making sense of experience. In *Funology: From Usability to Enjoyment*, ed. M. A. Blythe, K. Overbeeke, A. F. Monk, and P. C. Wright, 43–53. Dordrecht, the Netherlands: Kluwer Academic.

Weber, R., L. Ritterfeld, and K. Mathiak. 2006. Does playing violent video games induce aggression? Empirical evidence of a functional magnetic resonance imaging study. *Media Psychology* 8(1):39–60.

For more articles on emotion and game research, see http://www.xeodesign.com/whyweplaygames.

32 Motor Vehicle–Driver Interfaces

Paul A. Green

CONTENTS

32.1 INTRODUCTION

This chapter is written for professionals familiar with human–computer interaction (HCI), but not with the issues and considerations particular to motor vehicles. For non-HCI professionals, reading Chapters 1–29 of this text should provide the desired background. For automotive industry driver interface designers, this chapter should pull together information dispersed throughout the literature.

HCI is of interest to motor vehicle designers because of the rapid growth of driver information systems that utilize computers and communications, collectively referred to as "telematics." Included under the umbrella of telematics are navigation systems, cell phones, and so forth. Also important are "infotainment" systems, the combination of information systems (commonly navigation) and entertainment systems (music), a grouping that obviously overlaps with telematics. The interfaces for infotainment systems are far more complex than the two knobs, a dial, and five buttons of yore. These systems are being developed to (1) enhance public safety, (2) make transport more efficient (saving time and fuel), (3) make driving more enjoyable, and (4) make drivers more productive. It is with respect to these goals that systems are evaluated.

Although findings from research are important, this chapter emphasizes the resulting design documents and evaluation methods for driver interfaces to promote safety and ease of use. An underlying theme of this chapter is that the safety-critical and highly regulated nature of driving leads to significant departures from standard HCI practice, and to some methods, measures, and statistics that are unique to automotive applications. As this is a reference handbook, engineering practice receives more attention than scientific theory and, furthermore, given its technology focus, the design of traditional (noncomputer) motor vehicle driver interfaces (such as switches for headlights and windshield wipers) is not covered. For information on traditional interfaces, see Peacock and Karwowski (1993) or the latest edition of the Society of Automotive Engineers (SAE) Handbook (Society of Automotive Engineers 2010a,b). Readers interested in additional research literature on contemporary driver interfaces should see the proceedings from the biannual Driving Assessment conference, which occurs in the summer of odd numbered years. (For the 2009 Driving Assessment conference proceedings, see http://drivingassessment.uiowa.edu/node/17, retrieved May 22, 2010). A conference of increasing interest is the Automotive User Interfaces and Interactive Vehicular Applications Conference (see http://www.auto-ui.org, retrieved May 24, 2010).

32.1.1 What Kinds of New Features Are Likely in the Near Term?

To be able to design for systems of the future, one needs a sense of what they will be. Commonly, studies have used expert opinions to predict the future of automotive electronics, specifically telematics applications (e.g., Ribbens and Cole 1989; Underwood, Chen, and Ervin 1991; Underwood 1989, 1992; Richardson and Green 2000; Green et al. 2001). The most extensive information, however, is often contained in proprietary market surveys not for public distribution (e.g., Frost and Sullivan 2009).

Although the statistical accuracy of the proprietary market surveys is unknown, the published studies on the future are often too optimistic and, with surprising frequency, simply wrong. For example, Green et al. (2001) predicted that built-in cellular phones would be in 10% of all luxury cars in the 2004 model year. Interestingly, earlier research had rated cell phones as a low-priority feature. In part, this is because most studies are one-shot evaluations with no review of prior work to examine the basis for prior estimates and how they can be improved. Opportunities for authors to review their own estimates from previous years and generate more informed estimates are rare. Such estimates could be informed by utilizing quantitative historical sales data of vehicle various features to develop product diffusion models (Bass 1969, 2004; Mahajan, Muller, and Bass 1990, Wejnert 2002). (See also http://andorraweb.com/bass/, http://www.bassbasement.org/BassModel/.)

Although there are no firm statistics, recent research requests from manufacturers and publications in the open literature indicate there is considerable interest in audio texting/messaging and access to the web while driving for a wide range of purposes, as well as methods for music selection. A noteworthy development is real-time navigation systems for smart phones, in particular the iPhone. There is certainly the possibility that smart phones or tablets (e.g., iPad) could be the driver interface of the future. At this point, there is no evidence that these devices or applications for them are undergoing the safety evaluations required of in-vehicle devices used while driving.

32.1.2 Chapter Organization

How could one organize information on driver interfaces? In their classic paper on usability, Gould and Lewis (1985) identified three key principles to be followed when designing products for ease of use:

1. Early focus on users and tasks
2. Empirical measurement
3. Iterative design

These principles not only apply to office applications and web development, but automotive applications as well. In the automotive context, designers need to understand (1) who drives the vehicle (users), (2) what in-vehicle tasks they perform, (3) the driving task (the most important task), (4) task context, and (5) the consequence of task failures. These topics are the focus of the first part of this chapter.

Second, it is important to be able to measure driver and system performance (empirical measurement). That topic constitutes the second part of this chapter.

Surprisingly, there have been few reports of how iterative design is used in developing driver interfaces, though the approach is used. Complete attention to all three principles, however, is not common (Lee, Forlizzi, and Hudson 2008; Steinfeld and Tan 2000). Because a great deal of automotive design relies upon following design standards, that topic is the final focus of this chapter.

32.2 WHAT IS THE DRIVING CONTEXT IN WHICH USERS PERFORM TASKS?

Scott McNealy, CEO of Sun Microsystems, once said, “A car is nothing more than a Java technology-enabled browser with tires” (Kayl 2000). He is dead wrong. The author has never heard of anyone claiming, “A computer came out of nowhere, hit me, and vanished.” Yet police officers and insurance adjusters hear such claims about motor vehicle crashes every day. Likewise, the author knows of no one who has ever been killed as a consequence of operating a computer at their desk, but the loss of life associated with crashes arising from normal motor vehicle operation is huge.

According to the World Health Organization (WHO) (World Health Organization 2009), over 1.2 million people die in road traffic crashes each year, or almost 3300 per day, and somewhere between 20 and 50 million suffer injuries. WHO ranked traffic crashes as the ninth leading cause of death and the leading cause of death of adults aged 15–29. If the current trends continue, by the year 2030, traffic crashes will become the fifth largest cause of death after heart attacks, stroke, pneumonia, and lung diseases of various types.

Additional insights come from crash data for the United States, for which reliable, detailed crash statistics are available. In fact, the United States is probably the only country in the world for which its crash databases are available to anyone for free, which can lead those examining crash data to a U.S.-centric perspective of motor vehicle crash problems.

Analyses of U.S. crashes typically rely on three databases: (1) Fatality Analysis Reporting System (FARS), (2) National Automotive Sampling System (NASS), (3) General Estimates System (GES), and (4) Crashworthiness Data System (CDS). FARS (http://www.nhtsa.gov/people/ncsa/fars.html, retrieved June 1, 2010) is a database containing all fatal crashes in the United States. GES (www.nhtsa.gov/people/ncsa/nass_ges.html, retrieved June 1, 2010) is a nationally representative sample of police-reported crashes of all severities (including those that result in death, injury, or property damage). CDS (http://www.nhtsa.gov/PEOPLE/ncsa/nass_cds.html, retrieved June 1, 2010) is an annual probability sample of approximately 5000 police-reported crashes involving at least one passenger vehicle that was towed from the scene (out of a population of almost 3.4 million tow-away crashes). Minor crashes (involving property damage only) are not in CDS. CDS crashes are investigated in detail by specially trained teams of professionals who provide much more information than is given in police reports.

According to the U.S. Department of Transportation's 2008 annual traffic safety assessment (U.S. Department of Transportation 2009a), 37,261 people were killed in traffic crashes in the United States in that year. Of them, 14,587 were passenger car occupants; 10,764 were pickup truck occupants; 5,290 were on motorcycles; 4,378 were pedestrians; 677 were in larger trucks; 716 were bicyclists; and 188 were in other categories.

32.2.1 How Often and What Kinds of Crashes Are Associated with Telematics?

Crashes can occur for a wide variety of reasons (U.S. Department of Transportation 2008) and crashes are often attributable to multiple causes. It is widely accepted that some telematics tasks are distracting and that distraction can lead to crashes. The most recent analysis (U.S. Department of Transportation 2009b), based on FARS and GES concluded that distraction was reported for 11% of the drivers involved in fatal crashes, but those crashes were associated with 16% of all fatalities. Overall, the percentage of fatal crashes involving distraction has increased about 1% per year since 2004. Interestingly, the percentages of injury and property damage crashes have declined annually by a similar amount.

Additional details of distraction crashes appear in a previous analysis that utilized CDS (Wang, Knipling, and Goodman 1996) (Table 32.1). They found that distraction-related crashes primarily involved a single vehicle (41%), though rear-end crashes (moving, 10%; stopped, 22%) were also common. Intersection crashes represented another 18% of the total. Crashes tended to peak in the morning rush hour and, to a much lesser extent, in the evening rush. The overwhelming majority of the crashes occurred in good weather (86% clear, 10% rain, 4% snow/hail/sleet), and many occurred at lower speeds (40% at 0–35 mph, 40% at 40–50 mph, 17% at 55–60 mph, and 4% at over 65 mph). Thus, these data suggest that device test scenarios

TABLE 32.1
Distraction/Inattention Crashes by Crash Type

Crash Type Row (%) Column (%)	Sleepy	Distracted	Looked But Did Not See	Unknown	Attentive	Total
Single vehicle	5.8	18.1	0.2	31.8	44.1	100.0
	66.2	41.2	0.7	20.6	45.9	30.0
Rear-end/lead vehicle	12.7	21.3	3.4	48.3	14.3	100.0
moving	27.9	9.6	2.0	6.4	2.9	5.9
Rear-end/lead vehicle	*	23.9	11.4	52.6	11.8	100.0
stopped		21.9	13.8	14.1	4.9	12.1
Intersection/crossing	*	7.0	17.9	52.8	22.3	100.0
path		18.1	63.6	39.8	26.6	34.3
Lane change/merge	*	5.6	17.2	41.8	35.3	100.0
		1.6	6.7	3.4	4.6	3.8
Head-on	1.0	7.0	8.1	46.4	37.5	100.0
	1.7	2.2	3.5	4.3	5.4	4.2
Other	*	7.3	9.7	53.5	28.9	100.0
		5.4	9.7	11.4	9.7	9.7
Total crashes	2.6	13.2	9.7	45.7	28.8	100.0
	100.0	100.0	100.0	100.0	100.0	100.0

Source: Wang, J. S., R. R. Knipling, and M. J. Goodman. In *Association for the Advancement of Automotive Medicine 40th Annual Conference Proceedings*, 377–92. Association for the Advancement of Automotive Medicine. Des Plaines. IL. 1996.

*Too few cases for a stable estimate.

should emphasize situations in which single-vehicle crashes (often run-off-the-road), as well as those involving rear-end collisions into stopped vehicles, are likely. (See also Stutts, Reinfurt, Staplin, and Rodgman [2001] and Eby and Kostyniuk [2004].)

32.2.2 What Kinds of Trips Do People Make and Why?

Every 5–10 years, the U.S. Department of Transportation conducts the National Household Travel Survey to obtain travel data for the United States (Hu and Reuscher 2004), and many other countries conduct similar studies as well. (See http://www.dft.gov.uk/pgr/statistics/datatablespublications/personal/ and http://en.wikipedia.org/wiki/Travel_survey, retrieved May 31, 2010.) In the most recent U.S. data (2001), people in the United States are shown to have traveled an average of 14,500 miles per year, making four trips per day. They drove an average of 40 miles per day, with most of the miles (about 35) covered in a personal vehicle. Keep in mind that these are averages, and that public transit (including school buses) prevalent in urban areas accounts for only 2% of all trips. The travel situation is likely to be different for more urbanized countries (Japan, most of Europe), where public transit is more prevalent.

According to the 2001 data (Table 32.2), the most common reason for travel is family and personal business, which includes shopping, running errands, and dropping off and picking up others, accounting for almost half of the trips.

These and other data (on trip distances, travel speeds, time of day, etc.) in the National Household Travel Survey provide information on both the tasks and information needs that driver information systems should support and the conditions (road types, speed, weather, etc.) under which safety and usability should be assessed.

In contrast to the emerging understanding of the primary driving task, less is known about the real use of in-vehicle devices while driving, in particular, the frequency and duration of various tasks.

TABLE 32.2
Summary of Trip Purposes

Purpose	Person Trips (%)
Family and personal business	44.6
Work	14.8
Social and recreational	27.1
School and church	9.8
Work-related	2.9
Other	0.8

Source: Hu, P. S., and T. R. Reuscher. *Summary of Travel Trends: 2001 National Household Travel Survey*. U.S. Department of Transportation. Washington, DC. 2004. http://nhts.ornl.gov/2001/pub/STT.pdf (accessed April 3, 2007).

32.2.3 Who Are the Users?

Almost any adult has the potential to drive. To do so, they need only complete limited driver licensing requirements. Thus, in some ways, the driving population represents the population of candidate users for office computer systems. In the United States, within any age group, the percentages of men and women who are licensed are within 1% of being equal except for the elderly (Figure 32.1) (Highway Statistics 2008). Elderly women are sometimes more likely to drive because they are in better health. Notice that the percentage of the United States population that is licensed hits 80% at age 21 and increases with age, reaching a maximum at age 60–64, and then begins to decline. Thus, in designing in-vehicle systems for motor vehicles, few adults can be excluded, which differs from the design of office computer systems, where the emphasis is on the working population (generally less than 65 years old). Further, because of a wide age range, skill, and experience, significant differences in individual performance can be expected. For example, in the University of Michigan Transportation Research Institute (UMTRI) telematics studies, older drivers typically required one and a half to two times longer to complete tasks than younger drivers (Green 2001d). This fact, along with the requirement to design and test for the reasonable worst-case drivers, makes testing drivers over age 65 imperative. Of course, this is all for the United States, and in places where the vehicle market is growing rapidly such as China, there is a greater predominance of younger drivers.

In contrast to computer users, operators of motor vehicles must be licensed. In the United States, the process of becoming a licensed driver begins with obtaining a copy of the state driving manual and learning the state's traffic laws. Candidates must also pass vision tests (see http://www.lowvisioncare.com/visionlaws.htm) and take a test of rules of the road to obtain a learner's permit, often on or after their sixteenth birthdays. Consistent with the increasingly common practice of graduated driver licensing, learners can drive at restricted times with adult supervision. They must generally complete a driver's education class (which often includes gory crash movies designed to convince teenagers

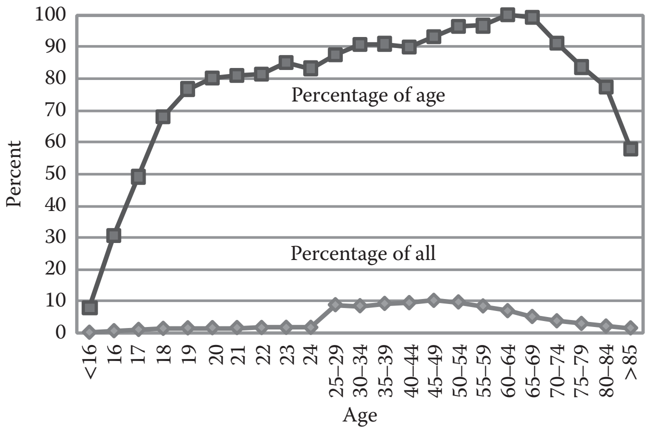


FIGURE 32.1 Distribution of driver age groups developed from U.S. Department of Transportation data.

not to drink and drive and to wear seat belts) and, after a few years, they pass an on-the-road test and obtain a license to drive. (For details, see http://www.iihs.org/laws/graduatedLicenseIntro.aspx, retrieved May 31, 2010.) The rationale for graduated licensing is to provide new drivers with more experience under less risky conditions. To put this in perspective, Mayhew, Simpson, and Pak (2003) show that crash rates per 10,000 novice drivers drop dramatically with time, being about 120, 100, and 70 after 1, 3, and 6 months of being licensed. Similarly, recognizing the increased risk of elderly drivers, some states have special renewal procedures for older drivers (http://www.iihs.org/laws/olderdrivers.aspx, retrieved May 31, 2010).

In the United States, obtaining a commercial driver's license, which is needed to drive buses, large trucks, and other vehicles, is a more complex process. Most candidates either obtain (1) on-the-job training, (2) training integrated into their lifestyle (using machinery on a farm), or (3) training at truck driving schools (Sloss and Green 2000). That population tends to be older than the working population as a whole, and is predominantly male.

Driver licensing practice varies from country to country (http://en.wikipedia.org/wiki/Driver's_license, retrieved May 31, 2010). Europe will have a common license in place in 2013 (http://en.wikipedia.org/wiki/European_driving_licence, retrieved May 31, 2010). In Japan, for example, the failure rate for the basic licensing exam is much higher than in the United States, and there is much greater use of special schools to train drivers. In some countries, obtaining a license may require minimal skill, training, or knowledge, and corruption of the licensing authority can be an issue (Bertand et al. 2008).

32.2.4 What Kinds of Vehicles Do People Drive (The Platform Question)?

For computers, people are concerned about the (1) brand, (2) amount of memory, (3) processor speed, (4) capacity of the hard drive, (5) type and version of operating system (Windows, Mac, or Linux), (6) type and version of browser, (7) the network connection speed, and so forth. The hardware and software of individual computers is in a state of flux, being constantly updated over a life span of often 3–5 years. Fortunately, the physical interface is fairly consistent—a QWERTY keyboard, mouse, and with increasing frequency, a 24-in. monitor for a desktop computer. The on-screen "desktop" is a more flexible space than the motor vehicle instrument panel, though there has been talk of personalizing instrument clusters, complete with personal photos.

Unlike personal computers, a motor vehicle is almost completely identified by its make, model, and year. Updates over an average 13-year life span are rare. (See http://www.bts.gov/publications/national_transportation_statistics/#chapter_2, Table 1-11 for the distribution of vehicles by year.) In most countries (at least where there is left-hand drive), many aspects of driving are fairly consistent: the (1) input devices (steering wheel, brake, and throttle), (2) method of operation, (3) location, and (4) primary displays (windshield and mirrors). In contrast, there is no consistency in the controls or displays for telematics interfaces. Furthermore, although new motor vehicle models are offered once per year, major changes typically occur every 5 years. Computer software and hardware model upgrades occur almost continually. Thus, the hardware lifecycles of the two contexts are quite different, and motor vehicle software is never updated unless there is a product recall.

As of 2008, there were 942 million vehicles in operation worldwide of which about 668 million were cars and 274 million were commercial vehicles (Wards Automotive Group 2009). Data on motorcycles and motorbikes was not provided. Of the world vehicle fleet, 296 million (31%) are in North America (mostly the United States), 238 million (25%) are in Western Europe. As of 2008, there were only 47 million vehicles in China, and 60% of those vehicles were commercial. That number is far less than Japan, with 74 million vehicles. However, there has been a huge change in the Chinese market for vehicles over the last few years. In 2008, there were 9.5 million vehicles produced in China (mostly for the Chinese market) and production has grown 1 million vehicles per year for the last several years. Sales of vehicles in China now exceed vehicle sales in the United States. However, relative to the United States and Europe, very little has been published about Chinese drivers or travel behavior.

Beyond the overall production and sales data, the vehicle sizes and types sold vary among countries, and even regions within a country. People on the east and west coast of the United States think of cars as the primary means of personal transportation. Yet, in other parts of the country, trucks, especially pickup trucks, predominate, and that is reflected in the U.S. sales shown in Table 32.3. Readers should keep in mind that these sales volumes represent a substantial change from a few years ago. Not only have total sales declined, but the sales of pickup trucks are half their totals from 5 years ago.

32.3 WHAT ARE SOME OF THE DEVICES OF CONCERN (AND SOLUTIONS)?

Driver distraction is a topic that has received considerable attention in government reports (e.g., Ranney 2008), has been a series in the *The New York Times* (topics.nytimes.com/top/news/…to_distraction/index.html, retrieved June 1, 2010), has become a cause on *The Oprah Winfrey Show* (a popular TV show in the United States), and even has its own government web site (www.distraction.gov). Although there are many potential distractions a driver might encounter, this chapter focuses only on those related to human–computer interfaces.

In addition to these external influences, the auto industry is strongly influenced by customer feedback on their products, especially as expressed in the J.D. Power Initial Quality Survey (measured at 90 days of ownership) and the Vehicle Dependability Survey (after 3 years of ownership). In the case of J.D. Power surveys, the focus is on the relative ratings of a particular manufacturer's products relative to the competition, not the absolute product ratings. The questions relating to navigation system characteristics receive considerable attention.

TABLE 32.3
Top 20 Selling Vehicles in the United States in 2009

Rank	Make	Model	Sales
1	**Ford**	**F-series**	**413,625**
2	Toyota	Camry	356,824
3	**Chevrolet**	**Silverado**	**316,544**
4	Toyota	Corolla	296,874
5	Honda	Accord	290,056
6	Honda	Civic	259,722
7	Nissan	Altima	203,568
8	Honda	CR-V	191,214
9	Ford	Fusion	180,671
10	**Dodge**	**Ram**	**177,268**
11	Ford	Escape	173,044
12	Chevrolet	Impala	165,565
13	Chevrolet	Malibu	161,568
14	Ford	Focus	160,433
15	Toyota	RAV4	149,088
16	Toyota	Prius	139,682
17	Hyundai	Sonata	120,028
18	GMC	Sierra	111,842
19	**Toyota**	**Tacoma**	**111,824**
20	Chevrolet	Cobalt	104,724

Source: http://www.autoblog.com/2010/01/06/top-20-best-selling-vehicles-of-2009/, retrieved May 27, 2010.
Note: Bolded terms are pickup trucks.

32.3.1 Cell Phone Problems

Because driver information systems are new, the data on device-related crashes is limited. This lack of crash data has hampered progress in understanding the risks of such devices, especially cell phones, which receive the most attention. At the time this chapter was written, many (but not all) states record whether cell phones are causal factors in crashes, with most just starting to collect this information recently. For a variety of reasons, identification of phone use as a causal factor is believed to be underreported. Eight states currently ban talking on a cell phone while driving, with 26 banning texting (http://www.iihs.org/laws/CellPhoneLaws.aspx, http://www.ghsa.org/html/stateinfo/laws/cellphone_laws.html, retrieved May 31, 2010). Bans on texting are becoming more common. Currently, 6% of all drivers are using cell phones at any moment (U.S. Department of Transportation 2009). However, given increased feature content (MP3 support, text messaging, broadcast TV), cell phone use could increase in the future.

There have been several studies that have examined vehicle crash risk related to cell phone use by drivers. The most often cited study of cellular phone use is Redelmeier and Tibshirani (1997; see also Redelmeier and Tibshirani 2001). They examined data for almost 700 drivers who were mobile phone users and were involved in motor vehicle crashes that resulted in substantial property damage. Each driver's mobile phone records for the day of the crash and the previous week were examined. Redelmeier and Tibshirani reported the risk of a crash was 4.3 times greater when a mobile phone was used than when it was not. Interestingly, handsfree units had a greater risk ratio (though not significant) than handheld units (5.9;1 vs. 3.9;1). Other data (Koushki, Ali, and Al-Saleh 1999; McEvoy et al. 2005; Violanti and Marshall 1996) suggested similar risk ratios. In contrast, data from Dingus and Klauer (2008) suggests a much lower risk. Meta-analyses and other reviews show drivers taking longer to respond to brake lights of lead vehicles, departing from the lane more often, and exhibiting other undesired characteristics while using a cell phone (Caird et al. 2006; Horrey and Wickens 2006; Mccartt, Hellinga, and Bratiman 2006; Collet, Guillot, and Petit 2010a,b). In contrast, more recent research shows no changes in crash risk or claims associated with cell phone use (Highway Loss Data Institute 2009; Farmer, Braitman, and Lund 2010).

In addition to the overall issue of crash risk, there has been some concern relating to manual dialing, answering the phone, conversation, or other tasks, and under what conditions these tasks occur. Useful data appear in Green, George, and Jacob (2003). One interesting data point comes from Nowakowski, Friedman, and Green (2001) who found that while driving in a simulator, most people answered a ringing phone in a few rings almost independently of the traffic situation. Answering the phone should not usually be more important than driving, but people behave otherwise out of habit.

Talking on the phone is different from talking to a passenger, especially an adult in the front seat. Often, that adult behaves as a codriver, looking both ways at intersections and checking the mirrors during lane changes, and speaking less in higher risk situations (Drews, Pasupathi, and Strayer 2004; Crundall et al. 2005). When drivers move their heads to scan an intersection or expressway entrance or exit, passengers often stop talking. People on the phone have no knowledge of the driving situation, and just keep talking. Admittedly, the situation can be complex, as talking to passengers can be distracting, but the most recent evidence suggests that speaking to callers (who are unaware of the driving situation) is worse (McEvoy, Stevenson, and Woodward 2007). The effects of conversation with others may very well depend upon a driver's age (teenagers are very susceptible to distraction) and the gender of the passenger (young men may drive less aggressively when young women are passengers) or called party, as well as the number of passengers. Technology could assist in reducing the scope of this problem, for example by providing cues to inform callers of the driving situation and alerting drivers of the call duration or their driving performance. In fact, a number of applications are appearing that block calls to phones when the phones are moving faster than a walking pace, primarily intended for parents who do not want their teens to talk on the phone or text while driving. Keep in mind that most of the time in the United States, vehicles have only one occupant—the driver.

32.3.2 Problems with Navigation Systems

The tasks associated with navigation system destination entry when performed using a visual-manual interface generated the initial concerns about driver distraction (e.g., Steinfeld et al. 1996; Nowakowski, Utsui, and Green 2000; Farber et al. 2000). These concerns continue today. (See http://www.cbc.ca/marketplace/2010/gps_distraction/main.html, GPS distraction, aired January 1, 2010, retrieved May 31, 2010.) More specifically, the concern is for entry of street addresses and intersections, and points of interest. For the street address, the driver is required to enter the city and the state, as well as the street name, building number, prefix (east, north, etc.) and suffix (street, road, drive, etc.). Not only does this require a significant number of keystrokes, which can take 40 seconds to 1 minute to enter, but often the driver may not know the complete address, in which case several alternative variations of the address need to be entered before the desired address is found. ("They live at 1015 Peachtree. Is that Peachtree Road, Avenue, Boulevard, Place, or something else?")

With points of interest, people often do not know in which category they will find their desired destination, so they end up exploring many dead ends. (e.g., is Cobo Hall in Detroit a civic center, a community center, or is it listed in some other category?)

32.3.3 Menu Interface Problems (Especially for Music Selection)

The iPod and other devices provide people with access to large music libraries. iPods with 4000 songs are not unusual, and that number will surely increase several fold in the future. The problem is that drivers try to retrieve songs, albums, and playlists while driving, and not just from one device, but potentially from an iPod or some other MP3 player, a USB drive, a hard disk drive, or a CD player, all of which can be connected at the same time. Drivers may need to go through multiple menus to get to the desired selection. Furthermore, selections can be shuffled, transferred, renamed, and sorted, all while driving. For current research on this topic, see Salvucci et al. (2007); Bayly, Young, and Regan (2008); Chisholm, Caird, and Lockhart (2008); Garay-Vega et al. (2010).

32.3.4 Web Access Problems

At this point, web access while driving is not very common, in part because access to the web in a moving vehicle is a recent occurrence. However, given the number of keystrokes required to enter a URL and the amount of information to be read on a web page, use of the web while driving in a manner typical of an office or home setting will be problematic.

32.3.5 Speech Interfaces—Are They the Future?

There are some who believe that speech interfaces are the solution to the problem of information access while driving. There is good evidence to suggest that speech interfaces can be less distracting than visual-manual interfaces (Tsimhoni, Smith, and Green 2004; Barón and Green 2006; Shutko, Mayer, Laansoo, and Tijerina 2009; Garay-Vega et al. 2010; Owens, McLaughlin, and Sudweeks 2010), though they are not without their problems (Chang et al. 2009). At this point, use of speech interfaces is uncommon because recognition performance is not very good. However, the considerable success of Ford Sync® and opportunities to process speech off board may change the situation.

32.3.6 Workload Managers—Are They the Future?

Given the concern for overload, one potential solution is to measure the primary task workload and then regulate the secondary tasks a driver can do at any moment using a workload manager (Michon 1993; Green 2000b, 2004; Hoedemaeker, de Ridder, and Janssen 2002; Piechulla et al. 2003). As initially conceived, such systems would use data from four sources: (1) the navigation system (such as lane width and radius of curvature), to assess the demands due to road geometry, (2) the adaptive cruise control system (headway and range rate to vehicles ahead), to assess traffic demands, (3) the traction control system, to assess road surface friction, and (4) the wipers, lights, and clock, to assess visibility. This information—along with information on the driver (e.g., age) and the specific visual, auditory, cognitive, and motor demands of each in-vehicle task—could be used to schedule the occurrence of in-vehicle tasks. Thus, when driving on a curving road in heavy traffic in a downpour, incoming mobile phone calls could be directed to an answering machine and the 30,000-mile maintenance reminder could be postponed. When the driving task demand is low, drivers could have access to a wide range of functions. Being able to reliably predict the momentary workload of driving, however, has proven to be very difficult.

As some vehicle engineers have realized, drivers are most likely to be overloaded when maneuvering or about to maneuver. Maneuvering includes (1) changing lanes, (2) merging onto an expressway, (3) turning at an intersection, (4) parking, (5) braking in response to a lead vehicle, (6) accelerating from a traffic light, and so forth. These situations are much easier to identify than the overload situations described earlier and, furthermore, are situations when drivers are less likely to desire additional tasks (such as responding to an incoming call) and would likely lead to much greater market acceptance of a workload manager (Eoh et al. 2006).

32.4 WHAT MEASURES AND STATISTICS OF SAFETY AND USABILITY ARE OF INTEREST? (THE EMPIRICAL MEASUREMENT ISSUE)

As was noted earlier, usability is difficult to achieve without empirical measurement. Superficial impressions suggest that the measurement of usability of office computer and web applications, and the measurement of the usability of driver interfaces are quite similar. In an office, one measures task completion time, errors, and ratings of ease of use. A typical usability lab has (1) a one-way mirror, (2) cameras, (3) video editing equipment, (4) audio mixers, and (5) at least two rooms, one for the subject and one for an experimenter. (See Chapter 53.)

TABLE 32.4
Some Driving-Specific Usability Measures

Category	Statistic
Lateral	Number of lane departures
	Mean and standard deviation of lane position
	Standard deviation of steering wheel angle
	Number of steering wheel reversals
	Time to line crossing
	Steering entropy
Longitudinal	Number of collisions
	Time to collision
	Standard deviation of gap (time or distance to lead vehicle)
	Mean and standard deviation of speed
	Speed drop during a task
	Heading entropy
	Number of braking events over some g threshold
Visual	Number of glances
	Mean glance duration
	Maximum glance duration
	Percentage of off-road glances greater than 2 s
	Total eyes-off-the-road time

Note: For definitions of some of these measures, see SAE J2499 (Society of Automotive Engineers 2010a).

In a typical laboratory for examining driver interfaces, the same measures may be obtained. However, other driving-specific measures, as listed in Table 32.4, may also be obtained, especially in driving simulators and on-the-road tests. (See Green [1995a,b,c].) A major problem is that most researchers do not define how these measures are collected, making studies difficult to compare, and when they are defined, definitions are inconsistent. For example, a lane departure could be determined with respect to the outside edge of the exterior mirror or the outside of a tire. The boundary could be with respect to the inside, middle, or outside edge of pavement marking. Together, these two factors alone alter the determination of a lane departure by a foot. Similarly, headway can be measured to the front or rear of a lead vehicle, a difference of about 15–16 ft. for a car and 55 ft. for a tractor trailer, all substantial differences. Ideally, in the future, authors will use the definitions in SAE J2499 (Society of Automotive Engineers 2010a,b) for consistency.

In addition to the standard measures of driving performance, a host of other performance measures are often collected in driving studies, which include: (1) ratings of workload (National Aeronautics and Space Administration Task Load Index [NASA TLX]), (2) measures of situation awareness, and (3) measures of object and event detection (pertaining to traffic). (See Roskam et al. [2002]; Tijerina et al. [2003]; Johansson et al. [2004].) In fact, simulator and on-road studies of telematics typically involve anywhere from 10–30 dependent measures, though in operational field tests the collection of several hundred measures in real time is not unusual (General Motors Corporation 2005; LeBlanc et al. 2006). Because the user is engaged in concurrent performance of manual control and information management tasks, the situation in some ways is similar to that described by Landry in the aerospace chapter. (See Chapter 33.)

A major challenge in assessing the safety and usability of telematics is dealing with the trade-offs that drivers naturally make. The impression is that when preoccupied with an in-vehicle task, drivers lose awareness of the driving context—that is, situation awareness. Drivers attempt to compensate by slowing down (to make driving easier), allowing for larger gaps and, if very preoccupied, paying less attention to steering (so lane variance and the number of lane departures increase). However, drivers can respond in strange ways. For example, if asked to use two different in-vehicle systems, one of which is not well designed, they might attempt to maintain equal performance on both: slow down more for the more difficult interface, but compensate by having better steering performance for the poorer interface. Assessment is difficult because the trade-off functions for all of these measures are unknown. One strategy used to overcome the trade-off problem is to minimize the opportunity for trade-offs. For example, this might include using cruise control to fix the speed (and in some cases, headway) and provide incentives and feedback to maintain driving performance, so only task time and errors might trade off.

32.5 HOW ARE DRIVER INTERFACES EVALUATED?

The assessment of mobile applications often occurs in contexts other than simulators and real roads as suggested above. Table 32.5 provides a summary of the contexts and their strengths and weaknesses. (See Green [1995a] for additional details.)

Over the last decade, enormous improvements have been made in the quality of the tools available for human factors evaluations of driver interfaces. These include

1. Reductions in the size of video cameras (to that of a postage stamp) and their cost
2. Significant improvements in the fidelity, along with significant reductions in the cost, of wide field-of-view driving simulators
3. Electronic innovations that allow packaging compact instrumentation systems in vehicles and that reduce power consumption (and heat generation)
4. GPS systems for precise tracking of vehicle location
5. Digital cellular phones for remote downloading of vehicle data

Two specific developments of note are the custom driving monitoring systems and low-cost vehicle data logging systems. The best known of the monitoring system is Drivecam (www.drivecam.com, retrieved May 31, 2010), which consists of two video cameras, one aimed forward and one facing the driver, mounted in front of the rearview mirror. Recording occurs when lateral or longitudinal acceleration exceeds some threshold. Drivecam is commonly used by vehicle fleet operators to monitor their drivers and parents to monitor their teenagers.

TABLE 32.5
Evaluation Contexts

	Method and Equipment	Advice and Comments
Focus groups	Groups of 8 to 12 people demographically similar to customers sit around a table and discuss a product or service guided by a facilitator. Camera is often behind one-way mirror. Generally done in multiple cities (one or two groups/city). Often conducted by a marketing firm.	Useful in getting ideas for product concepts, but not predictions of the safety or usability of new products because the products have not been used. Approach is sometimes used by manufacturers when a usability test might be more appropriate. Generally no quantitative data. Essential to report actual quotes from participants, not what the facilitator recalls.
Clinics	Customers in various cities are given the opportunity to experience a new product and its competition, often two or three vehicles, side by side. Customers say which product or feature they prefer.	Only exposes users to a limited number of options. Performance data often not collected. Approach is commonly used by industry. Because the results are highly proprietary, published studies are rare.
Part task simulation	A sample of users operate the device (e.g., computer simulation of a new radio) and user task times, errors, and comments are recorded. Test facility is not sophisticated.	Not done that often. Relatively less common now than in the past as simulators have improved and instrumented vehicles have become more common.
Driving simulator	Typically driving simulators are fixed base (no motion) and cost $25,000 to $250,000 each, but the major cost is for a full-time engineer (or engineers) or several graduate students to operate the simulator. Simulators at manufacturers tend to be in 1–3 million dollar range, though some are much more (e.g., Ford is about $10,000,000). One to five projectors with total 40–210 degree forward field of view, real vehicle cab, steering system with torque feedback, and realistic sound. Rear image may be projected or mirrors may be replaced with small LCDs. For advice, see Green et al. (2003); Green (2005).	Operation requires considerable experience. Simulator sickness is a major problem, especially for wide field of view and older driver. Each experiment requires construction of a test road/world and scripting the behavior of vehicles and pedestrians. Facility can require considerable space (e.g., 1000 ft.^{2}). Generally requires large number of fixed small (lipstick or smaller) cameras. Best-known vendors in the United States are Systems Technology Inc. ($5,000–$25,000, http://www.systemstech.com/content/view/23/39/), Realtime Technologies ($75,000–$150,000), www.simcreator.com, GlobalSim Corporation, ($75,000–$150,000, http://www.drivesafety.com/), and OKTAL in Europe (http://www.scaner2.com/php/).
Instrumented vehicle on test track or public roads	Production vehicle (usually a car and often in the past, a station wagon) is fitted with cameras aimed at driver, forward scene, instrument panel, and lane markings, and with sensors for steering wheel angle, brake pressure, speed, and headway. Eye fixation system may also be provided. System of interest is also installed. In a typical experiment, the driver is asked to follow a test route while a back seat experimenter operates the test equipment.	Typical cost is at least $100,000–$250,000 though some low-cost systems may utilize a single camcorder on a vertically mounted curtain rod in the back seat aimed at the instrument panel. Problem in the past has been finding enough space and power for the equipment, which has been solved by laptops and lipstick cameras. Commonly, instrumented research test vehicles are used, though there is increasing use of systems that can be installed in the subject's own vehicle.
Operational field test	Compact instrumentation is installed in a fleet of vehicles (10–50). Each vehicle is borrowed by a potential user for a week, a month, or even a year. Driving performance is surreptitiously recorded by the vehicle. Unlike an instrumented car, continuous video is not recorded. In addition to data recorded by the instrumented vehicle, GPS-determined location is also recorded. Vehicles are periodically polled for data (and data is automatically dumped) by an independent digital cellular phone. Test is confined to a single metropolitan area but tests in multiple areas are being planned.	Each test requires unique instrumentation. Tests are very expensive ($10,000,000 to $40,000,000) and can only be conducted with significant government support. Experiment generally lasts several years. Planning stage for experiments takes several years. At any given time, there may be only one operational field test in progress in the United States.

LCD = liquid crystal display; GPS = global positioning system.

There are several low-cost devices that plug into the onboard diagnostic II (OBDII) connector of the engine computer (often under the dash), to allow for easy recording of speed, engine RPM and other driving measures. The best-known product is Carchip (www.carchip.com, retrieved May 30, 2010). In some cases, these may be a substitute for CANalyzers, commercial data loggers that record all information on the controller–area network (CAN) bus, the bus containing all of the speed, throttle, brake, and other driving data. Those not familiar with driving research should be forewarned. CAN bus codes are manufacturer specific and proprietary, and obtaining access to the codebooks is not easy.

The data that most researchers often wish they had concerns where drivers are looking. Development of better, lower cost eye fixation systems is still needed, though the leading vendors of hardware for driving research, Seeing Machines, the manufacturers of FaceLab (http://seeingmachines.com/), and Smart Eye (http://www.smarteye.se), are making progress. Challenges include enhancing accuracy and ability to hold calibration, dealing with glasses (common for older drivers), and operation in bright sunlight.

32.6 WHAT DESIGN DOCUMENTS EXIST FOR DRIVER INTERFACES?

32.6.1 What Types of Documents Are There?

Although feedback from the empirical measurements just described is important, automotive design is strongly influenced by industry and government regulations, and international standards. For office applications, compliance with design guidelines, commonly called style guides, is achieved by providing application program interfaces in the operating system that assure that widgets such as windows, menus, and so forth all work in a consistent way. (See Chapter 41 by Mayhew and Follansee for related information on requirements specification.) For driver interfaces, publicly available, product-specific interface guidelines do not exist, but there are other types of important written materials (Green 2001a,b; Schindhelm et al. 2004). Automotive design documents fall into five general classes: (1) principles, (2) information reports, (3) guidelines, (4) recommended practices, and (5) standards. Principles give high-level recommendations for design and are similar to those found in office HCI applications, such as “design interfaces to minimize learning.”

Information report is a term used by the SAE to refer to a compilation of engineering reference data or educational material useful to the technical community. Information reports do not specify how something should be designed, but provide useful background information.

Guidelines give much more specific advice about how to design an interface element. For example, guideline 9 in Chapter 7 of Green (1995c, p. 41) stated, “Turn displays should show two turns in a row when the turns are in close proximity,” where close proximity means 0.1 miles apart or less. The impact of guidelines can depend on the issuing organization. For example, automotive design guidelines written by research organizations have no real authority. Guidelines written by the International Organization for Standardization (ISO), although technically voluntary, can become requirements, because in some countries, type approval (approval for sale) requires compliance with ISO guidelines. For vehicle models sold worldwide, global manufacturers find that building common vehicle systems that comply with ISO standards to be less costly than building noncompliant, country-specific systems. In Japan, the Japan Automobile Manufacturers Association (JAMA 2004) has a set of guidelines for navigation systems. Although theoretically voluntary, “requests” from the National Police Agency make the JAMA guidelines a requirement for all original equipment manufacturers (OEMs) in Japan.

Recommended practice, a term used by SAE, refers to specifications for a material, product design, or design or test procedure that are intended to guide standard engineering practice, often because they have not gained broad engineering acceptance. Commonly, a recommended practice is followed. A product liability action against a product (especially in the United States) is extraordinarily difficult to defend if the product design and evaluation deviate from recommended practice.

In some sense, a standard is a recommended practice that is a broadly accepted engineering practice, and must be followed. In the SAE context, “must” has an unusual connotation, because an SAE standard is technically voluntary because SAE has no enforcement powers. However, in a product liability context, a product not complying with an SAE standard is almost not defendable, and is unlikely to be purchased from a supplier by an OEM.

32.6.2 Non-ISO Documents

Table 32.6 provides a summary of the design document activities till date excluding those of the ISO (described later). As indicated in the table, the EU guidelines are quite brief and are merely statements of very general principles (e.g., those interfaces should be simple to operate), though they serve as the basis of the Alliance guidelines described later. Most of the early guidelines (Battelle, Harmonization of ATT Roadside and Driver Information in Europe [HARDIE], Technology Readiness Level [TRL], UMTRI), though they are quite detailed and still valuable, are rarely used because there are no requirements to use them. Documents that are followed are JAMA (as was described previously), the Alliance guidelines (because the Alliance members agreed to follow them, possibly to avoid government regulation), and SAE and ISO guidelines (both of which are accepted industry practice and are described in the Sections 32.6.3 to 32.6.5). For all of these guidelines, see http://www.umich.edu/~driving/guidelines/guidelines.html for unofficial electronic copies. (To avoid copyright problems, only draft versions have been posted for SAE and ISO documents.) Readers are cautioned that all of these documents are updated every few years, and they should verify that they are using the most recent version with the authoring organization.

TABLE 32.6
Major Non-ISO Telematics Guidelines and Recommended Practices

Common Document Name	Reference	Size (Pages)	Comments
Alliance guidelines	Alliance of Automobile Manufacturers (2006); version 3	90	Elaboration of the EU principles with details on the method and rationale, used by almost all manufacturers in the United States; key sections are principles 2.1 and 2.2, which still need development.
Battelle guidelines	Campbell, Carney, and Kantowitz (1997)	261	Voluminous document with references to interface design, emphasis on trucks. User interface has been said to have a Windows OS flavor, includes physical ergonomics information (e.g., legibility, control sizes) which are not included in the UMTRI guidelines.
EU guidelines	Commission of the European Communities (2007)	42	Mostly “motherhood” statements.
HARDIE guidelines	Ross et al. (1996)	480	Early set of European guidelines, less data than UMTRI or Battelle.
JAMA guidelines	Japan Automobile Manufacturers Association (2004)	15	First set of detailed design guidelines for driver interfaces. These guidelines are voluntary in Japan but followed by all OEMs there and sometimes by aftermarket suppliers. Some aspects are particular to Japan. Device location restrictions are important.
SAE J2364 (“15-second rule”)	Society of Automotive Engineers (2004a)	13	Specifies the maximum allowable task time and test procedures for navigation system tasks performed while driving for systems with visual displays and manual controls; also describes an interrupted vision (visual occlusion) method as well; see also SAE J2365.
SAE J2365 (SAE calculations)	Society of Automotive Engineers (2002)	23	Method to compute total task time for tasks not involving voice, used early in design to estimate compliance with J2364.
TRL guidelines	TRL Limited (2004)	56	Expansion of simple checklist.
UMTRI guidelines	Green, Levison, Paelke, and Serafin (1995)	111	First set of comprehensive design guidelines for the United States. Includes principles, general guidelines, and specific design criteria with an emphasis on navigation interfaces.

EU = European Union; UMTRI = University of Michigan, Transportation Research Institute; HARDIE = Harmonization of ATT Roadside and Driver Information in Europe; JAMA = Japan Automobile Manufacturers Association; OEM = original equipment manufacturer; SAE = Society of Automotive Engineers; TRL = Technology Readiness Level.

32.6.3 ISO Documents

Much of the ISO activity over the last decade has occurred under the auspices of ISO Technical Committee 22, Subcommittee 13 (ISO TC 22/SC 13-Ergonomics Applicable to Road Vehicles, in particular, Working Group 8 [WG8—Transport, Information, and Control Systems or TICS]; Green 2000a). WG8 has about 50 delegates from the major vehicle-producing nations, with the most 15 active members appearing at meetings held two to three times per year, usually in Europe.

Table 32.7 shows the standards and technical reports developed (or in progress) by WG8 that pertain to telematics. (For the complete list, go to the ISO TC 22/SC 12 portion of the ISO website [www.ISO.org].) Most of the standards can be quite general, sometimes not containing the detail found in the Battelle, HARDIE, or UMTRI guidelines. For a variety of reasons, ISO documents emphasize measurement methods and organization over specifications and safety limits. To promote international harmonization, national standards organizations, technical societies (e.g., SAE), and government organizations (e.g., U.S. Department of Transportation) often permit ISO standards to supersede their own standards, so ISO standards are very important.

Note: ISO documents follow a very well-defined, 3-year process through several stages (Preliminary Work Item [PWI], Committee Draft [CD], Draft International Standard [DIS], Final Draft International Standard [FDIS], and International Standard [IS]) as they are passed from the working group to the subcommittee to the technical committee, and finally, to the secretariat for review and approval. The major hurdles are the working group and subcommittee, where passage requires two-thirds of the nations participating. The emphasis of this process is on building a voluntary consensus. Some items that are more informational in nature become technical reports instead of standards. Because of the limited number of experts available, WG8 is very selective in adding items to its work program.

TABLE 32.7
ISO TC 22/SC 13/WG 8 Work Program

Effort	Summary	Status
Dialogue management principles and compliance procedures	Provides high-level ergonomic principles (compatibility with driving, consistency, simplicity, error tolerance, etc.) to be applied in the design of dialogues that take place between the driver of a road vehicle and an in-vehicle information system while the vehicle is in motion. Provides general directions on how to test for compliance.	Std 15005
Specifications and compliance procedures for in-vehicle auditory presentation	Provides requirements for auditory messages including signal levels, appropriateness, coding, and so on, along with compliance test procedures.	Std 15006
Measurement of driver visual behavior	Generally describes video-based equipment (cameras, recording procedures, etc.) and procedures (subject descriptions, experiment design parameters, tasks, performance measures, etc.) used to measure driver visual behavior.	Std 15007, part 1 and TS part 2
Legibility (visual presentation of information)	Provides requirements for character size, contrast, luminance, and so on, and specifies how they are to be measured.	Std 15008
Warning system messages	This state-of-the-art literature review (circa 2002) of warning systems covers topics such as alarm theories; the design of visual, auditory, and tactile warnings; redundancy; and so on. The report contains summaries of a significant number of studies.	TR 16352
Message priority	Provides two methods for determining a priority index for in-vehicle messages (e.g., navigation turn instruction, collision warning, low oil) presented to drivers while driving. For one method, priority is based on criticality (likelihood of injury if the event occurs and urgency [required response time]), determined on four-point scales by experts.	TS 16951
Suitability of TICS while driving	Generally describes a process for assessing whether a specific TICS, or a combination of TICS with other in-vehicle systems, is suitable for use while driving. It addresses a user-oriented TICS description and context of use, TICS task description and analysis, assessment, and documentation.	Std 17287
Occlusion method to assess distraction	Describes how the visual demand of a display can be assessed by periodically blocking (occluding) the driver's view of the display. Includes requirements for number and training of subjects, test hardware, viewing and occluded periods, and two metrics for data analysis.	Std 16673
Lane change test	Proposes a procedure for testing the demand of telematics devices using a PC-based driving simulator. Subjects perform a number of lane changes, some of which occur while using an in-vehicle device. The test is based on results of the ADAM project. Complements SAE and AAM procedures.	Proof Std 26022

TICS = Transport, Information, and Control Systems; ADAM = Advanced Driver Attention Metrics; SAE = Society of Automotive Engineers; AAM = Alliance of Automobile Manufacturers.

32.6.4 SAE J2364 (The 15-Second Rule)

Of the design documents in the literature, few have generated as much discussion as SAE Recommended Practice J2364, also known as "The 15-Second Total Task Time Rule" or "15-Second Rule." SAE J2364 establishes two procedures for determining if a navigation system–related data entry task involving visual displays and manual controls is excessive while driving (Society of Automotive Engineers 2004a,b; Green 1999c). The practice applies to OEM and aftermarket products. The practice does not and should not apply to voice interfaces or passenger operation because the task demands are fundamentally different. There is no reason why the requirements should not be applied to all visual-manual tasks, not just navigation, given its performance basis.

In developing this practice, criteria were selected to be related to crash risk, likely to lead to design improvements, and easy to measure. Some suggested this document should contain design criteria, such as a specification for a maximum number of items on a menu. To be nimble and allow for changes in technology, a performance-based practice was developed instead. Many measures were then proposed, with eyes-off-the-road time being most popular. The logic behind this measure is simple. If drivers are not looking at the road while driving, they are more likely to crash, with the likelihood of a crash increasing with increased eyes-off-the-road time. As shown in Green (1999d), this is reflected in the following equation:

$$\begin{aligned}&\text{No. of U.S. deaths in 1989}\\&\quad = \text{market.penetration.fraction} * [-0.133 + (0.0447 *\\&\qquad (\text{mean glance time})^{1.5} * (\text{No. of glances}) * (\text{frequency}))]\end{aligned}$$

where *market.penetration.fraction* is the fraction of vehicles with a system (10% -> 0.1), *mean glance time* is the number of seconds drivers looked away from the road, *no. of glances* is the number of times the device is looked at for each use sequence, and *frequency* is the number of use sequences per week.

As an example, a task—say, entering an address—could have a mean glance duration of 2.7 seconds, require an average of 27.5 glances per entry, and be performed twice per week.

Unfortunately, eyes-off-the-road time is time-consuming and expensive to measure, and requires specialized equipment and skilled personnel. Measurement also requires a fully functional system installed either in a simulator or test vehicle, something that is only available late in design. Invariably, problems are identified so close to production that few, if any, changes can be made. One of the key lessons from the Gould and Lewis design principles from HCI is that feedback from early on in the design process is critical.

A review of the literature at the time the practice was being developed (Green 1999d) found that task time while driving was highly correlated with eyes-off-the-road time, a finding that has been repeatedly observed. Looking at in-vehicle systems is time shared with driving, so the more the driver needs to look, the longer the task will take. Further, dynamic (on-the-road) task time and eyes-off-the-road time are correlated with static task time, the time to complete the task when the vehicle is parked (e.g., Green 1999d; Farber et al. 2000; Young et al. 2005). Static task time is easy to measure and can be done using prototypes available early in design.

The current SAE J2364 test procedure (Society of Automotive Engineers 2004a) requires that 10 subjects between the ages 45 and 65 be tested. Each subject completes five practice trials and three test trials in a parked vehicle, simulator, or laboratory mockup. The test cases to be examined (addresses for destination entry) are to be representative of what is planned for production. Interestingly, the choice of the address can have a marked impact on the task time.

In the static method, the subject performs the task, with the duration being from when the subject is told to start until the goal is achieved. Timing is continuous except for computational interruptions equal to or greater than 1.5 seconds, a time period during which the device is computing (e.g., a route). If feedback is provided to the driver, that period is excluded from the 15-second task time limit. The interface complies with J2364 if the mean of the log of the task times is less than the log of 15 seconds. (Logs were used to reduce the influence of long outliers.)

It must be emphasized that the 15-second limit is for a static test. On-the-road drivers will take 30%–50% longer overall, and furthermore alternate between looking inside the vehicle and looking at the road. The test procedure does not suggest that drivers can safely look away from the road continuously for 15 seconds.

Some have argued that use of static task time fails to identify interfaces requiring long glance durations. However, analysis of real products shows the primary risk is from tasks that take too long to complete. In fact, it is very difficult to think of driver tasks for navigation systems that have short total task times but very long glance durations. In real driving, people truncate glances to the interior when the glances become too long but tend to complete tasks, even if they are unacceptably long. In practice, eliminating tasks with long completion tasks (the worst tasks) also eliminates many of the tasks with long glance durations.

Nonetheless, J2364 provides an alternative method involving visual occlusion. In that method, either the subjects wear special liquid crystal display (LCD) glasses, or vision to the device is otherwise periodically interrupted, simulating looking back and forth between the road and the device. (Unlike driving, though, subjects do nothing in the occlusion interval.) The device is visible for 1.5 seconds and occluded for 1–2 seconds, with 1.5 seconds being recommended. Compliance is achieved if the sum of the log of the viewing times is less than the log of 20 seconds.

32.6.5 Alliance Principles

The Alliance of Automobile Manufacturers (AAM), the trade association of 11 major manufacturers of automobiles in the United States (GM, Ford, Toyota, Mercedes, etc.) has devoted considerable effort to developing design guidelines/principles (Alliance of Automobile Manufacturers 2006). Although the document scope states that it applies only to "telematic devices," (p. 7), it should apply to all types of driver interfaces (except speech), since the low-level tasks (reading displays, pressing buttons, etc.) and the manner in which those tasks interfere with driving are the same for all systems. Some, if not many, of the manufacturers verify that their products meet the Alliance guidelines and request that their suppliers document that products sold to them also comply.

The Alliance guidelines are a detailed elaboration of the 24 principles in the EU guidelines (e.g., Principle 1.1: "The system should be located … in accordance with relevant … standards.…" "No part … should obstruct any vehicle controls or displays …"), (p. 13). Those guidelines seem obvious at a high level, but defining precisely how they can be met is quite difficult. Each principle has four parts: (1) rationale (usually quite detailed), (2) criterion/criteria, (3) verification procedures, and (4) examples.

The most important principle is 2.1: "Systems with visual displays should be designed such that the driver can complete the desired task with sequential glances that are brief enough not to adversely affect driving" (p. 39). Two alternative sets of criteria are offered. Alternative A says that "single-glance durations generally should not exceed 2 seconds," and task completion "should not require more than 20 seconds of total glance time." (Notice use of the words "should," not shall.) There is debate as to what the percentile criterion for a single glance and the maximum task time should be (Go et al. 2006). Verification can be achieved by a visual occlusion procedure (1.5 seconds viewing time, 1.0 second occlusion time), or by monitoring eye fixations directly using either a camera aimed at the face or an eye fixation monitoring system in either a divided-attention or on-road test.

Alternative B requires that the number of lane departures "should" not exceed the number associated with a reference task such as manual radio tuning, and that cars following headway "should" not degrade under those conditions, either. The verification procedure is stipulated to be driving on a divided road (either real or simulated) at 45 mph or less in daylight, on dry pavement, with low to moderate traffic. Additional details are provided describing the location of the radio, the stations to choose among, what constitutes a trial, subject selection (equal numbers of men and women between the ages of 45 and 65), and so forth.

Although both procedures seem well described on the surface, additional details and constraints are needed to make those procedures repeatable. For example, the differences in performance between driving in "low" and "medium" traffic could be quite considerable and need to be quantified such as in Schweitzer and Green (2007). Also needed are additional specifications for acceptable levels of variation in speed and lane position, criteria that could be developed from the data in Lai's (2005) dissertation or from Jamson et al. (2008). For many tasks that involve database searches (of address lists, song files, etc.), compliance with the principle will depend on the size of the database and the subject's familiarity with it. Nonetheless, the principles represent a reasonable first step in developing a test protocol. On the other hand, on-road and simulator tests are extremely expensive and time-consuming, often impractical, and occur too late to have a useful impact. Therefore, the author strongly prefers simpler evaluation procedures such as J2364, and the task time estimates determined using SAE J2365 and Pettitt's calculation procedures described in the Section 32.7.2.

32.7 WHAT TOOLS AND ESTIMATION PROCEDURES CAN AID TELEMATICS DESIGN?

32.7.1 SAE J2365 Task Time Calculations

SAE J2365 (Green 1999a) was developed to allow designers and engineers to calculate completion times early in design, when the design is still a concept that can easily be modified. As with J2364, J2365 is for in-vehicle tasks involving visual displays and manual controls evaluated statically—that is, while parked (or in a bench-top simulation). SAE J2365 applies to both OEM and aftermarket equipment. Though intended for navigation systems, J2365 should provide reasonable estimates for most in-vehicle tasks involving manual controls and visual displays.

The calculation method is based on the Goals, Operators, Methods, and Selection rules (GOMS) model described by Card, Moran, and Newell (1980) with task time data from several sources. (See Kieras' Chapter 57 for background information on GOMS.) The keystroke data was drawn from UMTRI studies of the Siemens Ali-Scout navigation system (Steinfeld et al. 1996; Manes, Green, and Hunter 1998). Search times were based on Olson and Nilsen (1987–1988), and the mental time estimates were drawn from the Keystroke-Level Model (Card, Moran, and Newell 1983) and UMTRI Ali-Scout studies. Thus, the times shown in Table 32.8 have been tailored for the automotive context. (See also Nowakowski, Utsui, and Green [2000].)

The basic approach involves top-down, successive decomposition of a task. The analyst divides the task into logical steps. For each step, the analyst identifies the human and device task operators. Sometimes analysts get stuck using this approach because they are not sure how to divide a task into steps. In those cases, utilizing a bottom-up approach may overcome such roadblocks. For each goal, the analyst identifies the method used.

The analyst is advised to document each method using paragraph descriptions and then convert those descriptions into pseudo code. All steps are assumed to occur in series; multiple tasks cannot be completed at the same time. Furthermore, most drivers are assumed to use only visible, noncognitively loading shortcuts. Invisible shortcuts are likely to be used only by experts.

Next, the pseudo code task description is entered into an Excel spreadsheet. The analyst looks up the associated time for each operator listed in Table 32.8 and sums them to determine total task time. To assist in understanding the process, the practice provides a step-by-step example of entering a street address into a PathMaster/NeverLost navigation system, a popular U.S. product. For background on the calculation method, see Green (1999b).

The J2365 approach shares a number of assumptions, many of which are also shared with the basic GOMS model. For example, the model assumes error-free performance, which while not likely, can be adjusted for (say by increasing the computed value by 25%). Further, activities are assumed to be routine cognitive tasks, with users knowing each step and executing them in a serial manner. Again, adjustments in computed time can account for users sometimes forgetting what is next.

Though many of these assumptions are not true, adjustments can be made for them, and many times the adjustments are small. Furthermore, violations of assumptions tend to affect all interfaces equally, so decisions about which of several interfaces is best still hold. As a practical matter, the estimates are good enough for most engineering decisions. Readers should keep in mind that J2364 only requires the use of 10 subjects at most, so there is some error in those estimates. Those errors are likely to be as large as variability among analysts and among J2365 estimates.

TABLE 32.8
SAE J2365 Operator Times

			Time (seconds)	
Code	**Name**	**Operator Description**	**Young Drivers (18–30)**	**Older Drivers (55–60)**
Rn	Reach near	From steering wheel to other parts of the wheel, stalks, or pods	0.31	0.53
Rf	Reach far	From steering wheel to center console	0.45	0.77
C1	Cursor once	Press a cursor key once	0.80	1.36
C2	Cursor twice or more	Time/keystroke for the second and each successive cursor keystroke	0.40	0.68
L1	Letter or space once	Press a letter or space key once	1.00	1.70
L2	Letter or space twice or more	Time/keystroke for the second and each successive cursor keystroke	0.50	0.85
N1	Number once	Press the letter or space key once	0.90	1.53
N2	Number twice or more	Time/keystroke for the second and each successive number key	0.45	0.77
E	Enter	Press the Enter key	1.20	2.04
F	Function keys or shift	Press the function keys or Shift	1.20	2.04
S	Search	Search for something on the display	2.30	3.91
Rs	Response time of system scroll	Time to scroll one line	0.00	0.00
Rm	Response time of system new menu	Time for new menu to appear	0.50	0.50

Note: The keystroke times shown include the time to move between keys. System response times to display new menus may be empirically determined.

32.7.2 Pettitt's Occlusion Time Calculation

In his dissertation, Pettitt (2008) (see also Pettitt, Burnette, and Stevens [2007] for one of several summaries) developed a similar method for estimating occlusion task times. In the occlusion procedure (ISO 16673, SAE J2364), subjects are allowed to intermittently look at a display while performing a task, simulating looking back and forth between the road scene and an in-vehicle system. Typically, this is accomplished by having the subject wear goggles with LCD shutters, which alternate between the two states (viewable, occluded) every 1.5 seconds, the duration of a typical glance.

Pettitt's calculation is quite similar to that of SAE J2365, with additional rules to determine what can be done during occlusion intervals.

Rule 1: During the vision interval (assumed to be 1.5 seconds), the task progresses normally without interruption.

Rule 2: Operators that do not require vision (e.g., mental) begun during vision can continue during occlusion.

Rule 3: Only operators that do not require vision can begin during occlusion (e.g., a key can be pressed if the finger is already resting there).

Also, there are fewer operator times than in J2365 and except for reaching for the device (Rf = 0.31 seconds), the times are slightly different (M = 1.25 seconds, H = home/move to key = 0.62 seconds, K = press a key = 0.2 seconds).

Using these methods, Pettitt, Burnette, and Stevens reported estimates to be 10% high (range 2%–22%) for static task time (estimate for 6 tasks from 12 drivers) and 13% high (range 2%–12%) for total shutter open time in an occlusion procedure, a reasonable approximation for engineering estimates. Keep in mind that the data from drivers is not the true time, just another estimate.

32.7.3 In-Vehicle Information System Estimates

A more complex estimation procedure, the In-Vehicle Information System Design Evaluation and Model of Attention Demand (IVIS DEMAnD) model, described by Hankey et al. (2000a,b). The model, which runs under Windows 98 or Windows NT, allows analysts to calculate a wide range of performance characteristics for proposed user interfaces. (The CD can be obtained from the U.S. Department of Transportation, Federal Highway Administration, Turner-Fairbank Highway Research Center in McLean, Virginia.)

The model assumes there are five basic human resources: (1) visual input, (2) auditory input, (3) central processing, (4) manual output, and (5) speech output. Overload of any one of these resources will affect task performance. The more demanding an in-vehicle task, the greater the likelihood of a crash. The model does not include a haptic component because haptic displays are rare in contemporary vehicles. Although many have developed models of human performance that partition human resources more finely, a five-component model is sufficient for most in-vehicle analyses. The data in the model were based on four experiments: (1) Gallagher (2001), (2) Blanco (1999), (3) Biever (1999), and (4) research conducted by Westat, a consulting company. These experiments concentrated on reading visual displays while driving, though there was work on auditory information as well.

To use the model, analysts need to create a description of each task drivers perform. Generally, that involves selecting the task in question from a large library of tasks in the database, modifying an existing description, or creating one from scratch. Tasks are grouped into seven categories: (1) conventional, (2) search, (3) search-plan, (4) search-plan-interpret, (5) search-plan-compute, (6) search-compute, and (7) search-plan-interpret-compute. Analysts need to select the driver age category (or specify all ages), the traffic density, the road complexity, the reliance on symbols/labels, the location of the display, and other characteristics.

The output of the model includes about 20 parameters, such as (1) the expected number of glances, (2) total task time, (3) ratings of mental demand and frustration, (4) total task demand, and so forth. In addition, the model output specifies if design thresholds are exceeded. The model proposes two sets of thresholds: (1) yellow-line and (2) red-line. Yellow-line thresholds were sets of points at which there was a measurable degradation in driving performance ($p < .05$) from baseline driving in the research conducted to support model development. Red-line thresholds indicated that a composite group of surrogate safety measures of driving performance was substantially affected. The red-line values were determined primarily from the literature and expert opinion. Table 32.9 shows those thresholds.

Quite frankly, IVIS DEMAnD is an orphan model that is not used, though it is certainly interesting. Unfortunately, the IVIS DEMAnD source code has not been widely distributed and the results from IVIS DEMAnD modeling have not been independently validated. Currently, the development of many other human performance models is better funded and they are used more often (GOMS, Adaptive Control of Thought—Rational [ACT-R], Executive-Process Interactive Control [EPIC], State, Operator and Result [SOAR], Operator Model Architecture [OMAR], etc.).

In particular, two modeling tools have garnered more interest than others, Distract-R (Salvucci et al. 2005; Salvucci 2009; www.cs.drexel.edu/~salvucci/distract-r/, retrieved May 31, 2010) and Cogtool (Teo and John 2008, http://cogtool.hcii.cs.cmu.edu/, retrieved May 31, 2010). Distract-R is a Macintosh-based application that allows for prototyping and evaluation of driver interfaces. Interfaces are prototypes from a pallet that includes displays, buttons, microphones, and speakers. Tasks are created by demonstration and described in an ACT simple framework. The key actions are: press a button (100 milliseconds preparation time plus a movement time determined by Fitts' Law, move a hand from the steering wheel (610 milliseconds), look at and encode information (150 milliseconds), begin to speak (300 milliseconds), begin to listen (300 milliseconds), rotate a knob (2 ms/degree), and think (1250 milliseconds). Also important in Distract-R is an ACT-R driver model that realistically emulates how drivers negotiate curves, change lanes, as well as where they look. Distract-R also has features that allows for the specification of driver characteristics (young and old), driver steering style, various driving scenarios (straight and curved roads, if a lead vehicle brakes), and then allows those simulations to be run and viewed. Running a simulation involves the ACT-R code to execute the secondary task, the ACT-R code to drive, plus addition code to coordinate the two tasks. Notably, there has been a modest level of effort to validate Distract-R, including one effort in collaboration with Ford.

Cogtool is a general-purpose user-interface prototyping tool that automatically generates the ACT-R code listing describing the human performance of a task being examined. There are versions of Cogtool for both Macs and PCs. Cogtool in many ways resembles Distract-R, though it uses a storyboard to create the interface. However, Cogtool has features for generating a wide variety of widgets (e.g., cascading menus) that do not appear to yet be supported in Distract-R. In contrast, Cogtool only supports the determination of device task times and does not have the driving features of Distract-R. There is a fair amount of support information for Cogtool, including instructional videos.

TABLE 32.9
In-Vehicle Information System Yellow- and Red-Line Thresholds

Measure	Affected (Yellow)	Substantially Affected (Red)
Single glance time	1.6 seconds	2.0 seconds
Number of glances	6 glances	10 glances
Total visual task time	7 seconds	15 seconds
Total task time	12 seconds	25 seconds

Source: Hankey, J. M., T. A. Dingus, R. J. Hanowski, W. W. Wierwille, and C. Andrews. *In-vehicle Information Systems Behavioral Model and Design Support Final Report,* Technical Report FHWA-RD-00-135. U.S. Department of Transportation, Federal Highway Administration. McClean. VA. 2000, http://www.tfhrc.gov/humanfac/00-135.pdf (accessed April 3, 2007).

32.8 CLOSING THOUGHTS

This chapter makes the following key points:

1. Driving is quite different from sitting at a desk in an office because of the concern for crash risk. People die while driving, lots of them.
2. Nonetheless, driver interface design should follow the same golden Gould and Lewis principles used to design ordinary office applications—(1) early focus on users and tasks, (2) empirical measurement, and (3) iterative design.
3. Drivers can range widely in age, and at least in the United States, almost all adults drive. Outside the United States, there is little literature on how drivers are licensed, and in some cases, there are doubts about how strictly licensing procedures are followed. Thus, there are significant gaps in knowledge of the user population, drivers. Nonetheless, it is important to include older drivers (over age 65), the least capable users, in safety and usability tests. Demographic data on drivers outside the United States is limited, with data on Chinese drivers being a special need.
4. The largest market for motor vehicles is now China. In the United States, which has been the largest market for a long time, the number of cars and trucks sold has been often equal, in part because of the popularity of pickup trucks and sport-utility vehicles. Worldwide, commercial vehicles continue to make up a large fraction of the world's fleet.
5. Trips are made for a wide variety of purposes that need to be considered in assessing driver interfaces.
6. A large number of new applications have appeared and will continue to appear in vehicles over the next few years, especially on phones and tablets for navigation, audio text messaging, and web access. New applications could substantially reshape the driver's task, providing the driver with a flood of information, especially if the driving task is more automated. The current major concerns are tasks related to cell phone use (especially conversation), navigation destination entry, and music selection.
7. This flood of information has the potential to distract drivers from driving. Distraction is associated with specific types of crashes (single vehicle run off road, rear end) that are most common under generally good driving conditions.
8. Workload managers and speech interfaces may be a long-term alternative to legislation to reduce distraction that leads to crashes.
9. In assessing safety and usability, a wide variety of statistics describing longitudinal and lateral control, per SAE 2499, are used in addition to task completion time, errors, and subjective ratings of ease of use. It is not unusual for there to be 30–50 dependent measures in a driving simulator study, and hundreds in an on-the-road study. A major challenge in motor vehicle interface evaluation is dealing with performance trade-offs between measures.
10. Over the last few years, there have been significant advances in driving simulators and instrumented vehicles that have improved their quality and reduced their cost for safety and usability evaluations. However, the cost of these systems may be out of the range of most laboratories, especially those in academic settings, though low-cost options for on-the-road evaluation (DriveCam, Carchip) are becoming more common.
11. The key design and evaluation documents are the Alliance guidelines, SAE J2364 and J2365, and a collection of standards from ISO Technical Committee 22, Subcommittee 13. In Japan, the JAMA guidelines are important.
12. SAE J2365, Pettitt's method, CogTool, and Distract-R can and have been used to predict task performance time, a simple measure for evaluating telematics safety and usability. The IVIS DEMAnD model can also serve that purpose, but it is not used.

Thus, while the HCI literature provides a framework for test methods and evaluation, a great deal is specific to the motor vehicle context because of the safety-critical nature of the context and the timesharing not found in office activities. To meet the needs of the future, the cost of the methods needs to be reduced, and reliable tools, especially for recording eye fixations, are needed. Significant research is needed to support the development of driver performance models (and workload managers) and understand how drivers use real driver interfaces.

REFERENCES

Alliance of Automobile Manufacturers. 2006. *Statement of Principles on Human-Machine Interfaces (HMI) for In-Vehicle Information and Communication Systems* (version 3). Washington, DC: Alliance of Automobile Manufacturers. http://www.umich.edu/,driving/guidelines/guidelines.html (accessed April 3, 2007).

Barón, A., and P. Green. 2006. *Safety and Usability of Speech Interfaces for In-Vehicle Tasks while Driving: A Brief Literature Review*. Technical Report UMTRI-2006-5. Ann Arbor, MI: University of Michigan Transportation Research Institute.

Bass, F. 1969. A new product growth model for consumer durables. *Manag Sci* 15(5):215–22.

Bass, F. 2004. Comments on "A new product growth for model consumer durables." *Manag Sci* 50(12 Suppl):1833–40.

Bayly, M., K. L. Young, and M. A. Regan. 2008. Sources of distraction inside the vehicle and their effects on driving performance. In *Driver Distraction: Theory, Effects, and Mitigation*, eds. M. A. Regan, J. D. Lee, and K. Young, 192–210. Boca Raton, FL: CRC Press.

Bertand, M., S. Djankov, R. Hanna, and S. Mullainathan. 2008. Corruption in driving licensing process in Dehli. *Econ Polit Wkly* (February):71–6. http://www.povertyactionlab.org/sites/default/files/publications/155%20%28b%29%20Corruption%20in%20Drivers%20Licence.pdf (accessed May 31, 2010).

Biever, W. J. 1999. Auditory based supplemental information processing demand effects on driving performance. Unpublished master's thesis, Virginia Polytechnic Institute and State University, Blacksburg.

Blanco, M. 1999. Effects of in-vehicle information systems (IVIS) tasks on the information processing demands of a commercial vehicle operations (CVO) driver. Unpublished master's thesis, Virginia Polytechnic Institute and State University, Blacksburg.

Caird, J. K., C. T. Scialfa, G. Ho, and A. Smiley. 2006. A meta-analysis of driving performance and crash risk associated with the use of cellular telephones while driving. In *Proceedings of the Third International Driving Symposium on Human factors in Driver Assessment, Training and Vehicle Design*. Iowa City, IA: University of Iowa.

Campbell, J. L., C. Carney, and B. H. Kantowitz. 1997. *Human Factors Design Guidelines for Advanced Traveler Information Systems (ATIS) and Commercial Vehicle Operations (CVO).* Technical Report FHWA-RD-98-057. Washington, DC: U.S. Department of Transportation, Federal Highway Administration.

Card, S. K., T. P. Moran, and A. Newell. 1980. The keystroke-level model for user performance time with interactive systems. *Commun ACM* 7:396–410.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Chang, J. C., A. Lien, B. Lathrop, and H. Hees. 2009. Usability evaluation of a Volkswagen group in-vehicle speech system. In *Proceedings of the First International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI2009)*. Essen, Germany.

Chisholm, S. L., J. K. Caird, and J. Lockhart. 2008. The effects of practice with MP3 players on driving performance. *Accid Anal Prev* 40(2):704–13.

Collet, C., A. Guillot, and C. Petit. 2010a. Phoning while driving I: A review of epidemiological, psychological, behavioural and physiological studies. *Ergonomics* 53(5):589–601.

Collet, C., A. Guillot, and C. Petit. 2010b. Phoning while driving II: A review of driving conditions influence. *Ergonomics* 53(5):602–16.

Commission of the European Communities. 2007. *European Statement of Principles on the Design of Human Machine Interaction*. Brussels, Belgium. http://eur-lex.europa.eu/LexUriServ/site/en/oj/2007/l_032/l_03220070206en02000241.pdf (accessed May 28, 2010).

Crundall, D., M. Bains, P. Chapman, and G. Underwood. 2005. Regulating conversation during driving: A problem for mobile phones? *Transp Res Part F: Traffic Psychol Behav* 8:197–211.

Dingus, T. A., and S. G. Klauer. 2008. *The Relative Risks of Secondary Task Induced Driver Distraction*. SAE paper 2008-21-0001. Warrendale, PA: Society of Automotive Engineers.

Drews, F. A., M. Pasupathi, and D. L. Strayer. 2004. Passenger and cell-phone conversations in simulated driving. In *Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting*, 2210–2. Santa Monica, CA: Human Factors and Ergonomics Society.

Driving Assessment. 2005. *Proceedings of the Third International Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*. Iowa City, IA: University of Iowa. http://ppc.uiowa.edu/driving-assessment/2005/final/index.htm (accessed April 3, 2007).

Eby, D. W., and L. P. Kostyniuk. 2004. *Distracted Driving Scenarios: A Synthesis of Literature, 2001 Crashworthiness Data System (CDS) Data, and Expert Feedback.* SAVE-IT project Technical Report, Task 1. Washington, DC: U.S. Department of Transportation. http://www.volpe.dot.gov/hf/roadway/saveit/docs/dec04/finalrep_1.pdf (accessed April 3, 2007).

Eoh, H., P. A. Green, J. Schweitzer, and E. Hegedus. 2006. *Driving Performance Analysis of the ACAS FOT: Data and Recommendations for a Driving Workload Manager*. Technical Report UMTRI-2006-18. Ann Arbor, MI: University of Michigan Transportation Research Institute.

Farber, E., M. Blanco, J. P. Foley, R. Curry, J. Greenberg, and C. Serafin. 2000. Surrogate measures of visual demand while driving. In *Proceedings of the IEA/HFES 2000 Congress* [CD-ROM]. Santa Monica, CA: Human Factors and Ergonomics Society.

Farmer, C. M., K. A. Braitman, and A. K. Lund. 2010. *Cell Phone Use While Driving and Attributable Crash Risk*. Arlington, VA: Insurance Institute for Highway Safety.

Frost and Sullivan. 2009. *Analysis of North American market of Advanced Driver Assistance Systems–ACC, LDW, BSD, Night Vision and Park Assist Systems*. Document N6450-18. Palo Alto, CA: Frost and Sullivan.

Gallagher, J. P. 2001. An assessment of the attention demand associated with the processing of information for in-vehicle information systems (IVIS). Unpublished master's thesis, Virginia Polytechnic Institute and State University, Blacksburg.

Garay-Vega, L., A. K. Pradhan, G. Weinberg, B. Schmidt-Nielsen, B. Harsham, Y. Shen, G. Divekar, M. Romoser, M. Knodler, and D. L. Fisher. 2010. Evaluation of different speech and touch interfaces to in-vehicle music retrieval systems. *Accid Anal Prev* 42(3):913–20.

General Motors Corporation. 2005. *Automotive Collision Avoidance System Field Operational Test (ACAS FOT) final program report.* DOT HS 809 886. Washington, DC: U.S. Department of Transportation, National Highway Traffic Safety Administration.

Go, E., A. Morton, J. Famewo, and H. Angel. 2006. *Final Report: Evaluation of Industry Safety Principles for In-Vehicle Information and Communication Systems*. Ottawa, Canada: Transport Canada.

Gould, J. D., and C. Lewis. 1985. Designing for usability: Key principles and what designers think. *Commun ACM* 28(3):300–11.

Green, P. 1995a. Automotive techniques. In *Research Techniques in Human Engineering*, ed. J. Weimer, 2nd ed., 165–208. New York: Prentice-Hall.

Green, P. 1995b. *Measures and Methods Used to Assess the Safety and Usability of Driver Information Systems.* Technical Report FHWA-RD-94-088. McLean, VA: U.S. Department of Transportation, Federal Highway Administration.

Green, P. 1995c. *Suggested Procedures and Acceptance Limits for Assessing the Safety and Ease of Use of Driver Information Systems.* Technical Report FHWA-RD-94-089. McLean, VA: U.S. Department of Transportation, Federal Highway Administration.

Green, P. 1999a. Estimating compliance with the 15-second rule for driver-interface usability and safety. In *Proceedings of the Human Factors and Ergonomics Society 43rd Annual Meeting* [CD-ROM]. Santa Monica, CA: Human Factors and Ergonomics Society.

Green, P. 1999b. *Navigation System Data Entry: Estimation of Task Times*. Technical Report UMTRI-99-17. Ann Arbor, MI: University of Michigan, Transportation Research Institute.

Green, P. 1999c. The 15-second rule for driver information systems. In *ITS America Ninth Annual Meeting Conference Proceedings* [CD-ROM]. Washington, DC: Intelligent Transportation Society of America.

Green, P. 1999d. *Visual and Task Demands of Driver Information Systems*. Technical Report UMTRI-98-16. Ann Arbor, MI: University of Michigan Transportation Research Institute.

Green, P. 2000a. The human interface for ITS display and control systems: developing international standards to promote safety and usability. Paper presented at the International Workshop on ITS Human Interface, Utsu, Japan.

Green, P. 2000b. Crashes induced by driver information systems and what can be done to reduce them. SAE paper 2000-01-C008. In *Convergence 2000 Conference Proceedings*, 27–36. Warrendale, PA: Society of Automotive Engineers.

Green, P. 2001a. Safeguards for on-board wireless communications. Paper presented at the Second Annual Plastics in Automotive Safety Conference, Troy, Michigan.

Green, P. 2001b. *Synopsis of Driver Interface Standards and Guidelines for Telematics as of mid-2001*. Technical Report UMTRI-2001-23. Ann Arbor, MI: University of Michigan, Transportation Research Institute.

Green, P. 2001c. Variations in task performance between younger and older drivers: UMTRI research on telematics. Paper presented at the Association for the Advancement of Automotive Medicine Conference on Aging and Driving, Southfield, MI.

Green, P. 2004. Driver distraction, telematics design, and workload managers: safety issues and solutions. SAE paper 2004-21-0022. In *Proceedings of the 2004 International Congress on Transportation Electronics*, 165–80. Warrendale, PA: Society of Automotive Engineers.

Green, P. 2005. How driving simulator data quality can be improved. In *Proceedings of the Driving Simulation Conference North America 2005* [CD-ROM]. November, 2005, Orlando, FL. http://www.umich.edu/~driving/publications.html (accessed April 3, 2007).

Green, P. 2008a. Driver interface safety and usability standards: An overview. In *Driver Distraction: Theory, Effects, and Mitigation*, Chap 24, eds. M. A. Regan, J. D. Lee, and K. L. Young, 445–64. Boca Raton, FL: CRC Press.

Green, P. 2008b. Driver interface/HMI standards to minimize driver distraction/Overload. SAE paper 2008-21-2002. In *Convergence 2008 Conference Proceedings*, Detroit, MI. Warrendale, PA: Society of Automotic Engineers.

Green, P., C. Nowakowski, K. Mayer, and O. Tsimhoni. 2003. Audio-visual system design recommendations from experience with the UMTRI driving simulator. In *Proceedings of the Driving Simulator Conference North America 2003* [CD-ROM]. Dearborn, MI. http://www.umich.edu/~driving/publications.html (accessed April 3, 2007).

Green, P., J. George, and R. Jacob. 2003. *What Constitutes a Typical Cell Phone Call*. Technical Report UMTRI-2003-38. Ann Arbor, MI: The University of Michigan Transportation Research Institute.

Green, P., M. Flynn, G. Vanderhagen, J. Ziomek, E. Ullman, and K. Mayer. 2001. *Automotive Industry of Trends in Electronics: Year 2000 Survey of Senior Executives*. Technical Report UMTRI-2001-15. Ann Arbor, MI, University of Michigan, Transportation Research Institute.

Green, P., W. Levison, G. Paelke, and C. Serafin. 1995. *Preliminary Human Factors Guidelines for Driver Information Systems*. Technical Report FHWA-RD-94-087. McLean, VA: U.S. Department of Transportation, Federal Highway Administration.

Hankey, J. M., T. A. Dingus, R. J. Hanowski, W. W. Wierwille, and C. Andrews. 2000a. *In-vehicle Information Systems Behavioral Model and Design Support Final Report*. Technical Report FHWA-RD-00-135. McClean, VA: U.S. Department of Transportation, Federal Highway Administration. http://www.tfhrc.gov/humanfac/00-135.pdf (accessed April 3, 2007).

Hankey, J. M., T. A. Dingus, R. J. Hanowski, W. W. Wierwille, and C. Andrews. 2000b. *In-vehicle Information Systems Behavioral Model and Design Support: IVIS Demand Prototype Software User's Manual*. Technical Report FHWA-RD-00-136. McClean, VA: U.S. Department of Transportation, Federal Highway Administration. http://www.tfhrc.gov/humanfac/00-136.pdf (accessed April 3, 2007).

Highway Loss Data Institute. 2009. *Hand-held Cell Phone Laws and Collision Claim Frequencies*. Arlington, VA: Highway Loss Data Institute.

Highway Statistics. 2008. *Distribution of Licensed Drivers-2008 by Sex and Percentage in Each Age Group and Relation to Population*. U.S. Department of Transportation, Federal Highway Administration. http://www.fhwa.dot.gov/policyinformation/statistics/2008/dl20.cfm (accessed June 1, 2010).

Hoedemaeker, M., S. N. de Ridder, and W. H. Janssen. 2002. *Review of European Human Factors Research on Adaptive Interface Technologies for Automobiles*. TNO report TM-02-C031. Soesterberg, The Netherlands: TNO Institute for Perception.

Horrey, W. J., and C. D. Wickens. 2006. Examining the impact of cell phone conversations on driving using meta-analytic techniques. *Hum Fact* 48(1):196–205.

Hu, P. S., and T. R. Reuscher. 2004. *Summary of Travel Trends: 2001 National Household Travel Survey*. Washington, DC: U.S. Department of Transportation. http://nhts.ornl.gov/2001/pub/STT.pdf (accessed April 3, 2007).

Insurance Institute for Highway Safety (IIHS). 2006. *US Driver Licensing Procedures for Older Drivers*. Arlington, VA. http://www.iihs.org/laws/state_laws/older_drivers.html (accessed April 3, 2007).

International Organization of Standardization. 2001. *Road Vehicles—Measurement of Driver Visual Behaviour with Respect to Transport Information and Control Systems—Part 2: Equipment and Procedures*. ISO Technical Specification 15007-2: 2001. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2002a. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Dialogue Management Principles and Compliance Procedures*. ISO Standard ISO 15005: 2002. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2002b. *Road Vehicles—Measurement of Driver Visual Behaviour with Respect to Transport Information and Control Systems—Part 1: Definitions and Parameters*. ISO Standard 15007-1: 2002. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2003a. *How are ISO Standards Developed?* Geneva, Switzerland. http://www.iso.ch/iso/en/stdsdevelopment/whowhenhow/how.html (accessed April 3, 2007).

International Organization of Standardization. 2003b. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Procedure for Assessing Suitability for Use While Driving*. ISO Standard 17287:2003. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2004a. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems (TICS)—Procedures for Determining Priority of On-Board Messages Presented to Drivers*. ISO Trial Standard 16951:2004. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2004b. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Specifications and Compliance Procedures for In-Vehicle Auditory Presentation*. ISO Standard 15006:2004. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2005. *Road Vehicles—Ergonomic Aspects of In-Vehicle Presentation for Transport Information and Controls Systems—Warning Systems*. ISO Technical Report 16352:2005. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2007a. *List of Technical Committees: TC 22/SC 13 Ergonomics Applicable To Road Vehicles*. Geneva, Switzerland. http://www.iso.org/iso/en/CatalogueListPage.CatalogueList?COMMID=869&scopelist=PROGRAMME (accessed April 4, 2007).

International Organization of Standardization. 2007b. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Occlusion Method to Assess Visual Distraction Due to the Use of In-Vehicle Information and Communication Systems*. ISO Draft Standard 16673. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2009. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Specifications and Compliance Procedures for In-Vehicle Visual Presentation*. ISO Standard 15008:2009. Geneva, Switzerland: International Organization of Standardization.

International Organization of Standardization. 2010. *Road Vehicles—Ergonomic Aspects of Transport Information and Control Systems—Simulated Lane Change Test to Assess In-Vehicle Secondary Task Demand*. ISO Proof Draft Standard 26022. Geneva, Switzerland: International Organization of Standardization.

Jamson, S., M. Wardman, R. Batley, and O. Carsten. 2008. Developing a driving safety index using a Delphi state preference experiment. *Accid Anal Prev* 40:435–42.

Japan Automobile Manufacturers Association. 2004. *JAMA Guideline for In-Vehicle Display Systems, version 3.0*. Tokyo, Japan: Japan Automobile Manufacturers Association. http://www.jama.or.jp/safe/guideline/pdf/jama_guideline_v30_en.pdf (accessed April 3, 2007).

Johansson, E., J. Engstrom, C. Cherri, E. Nodari, A. Toffetti, R. Schindhelm, et al. 2004. *Review of Existing Techniques and Metrics for IVIS and ADAS Assessment.* AIDE deliverable 2.2.1. Brussels, Belgium: European Union.

Kayl, K. 2000. *The Networked Car: Where the Rubber Meets the Road*. http://sun.systemnews.com/articles/32/1/ja/2746 (accessed April 5, 2007).

Koushki, P. A., S. Y. Ali, and O. I. Al-Saleh. 1999. Driving and using mobile phones: Impacts on road accidents. *Transp Res Rec* 1694:27–33.

Lai, F. C. H. 2005. Driver attentional demand to dual task performance. Unpublished doctoral dissertation, University of Leeds, Institute for Transport Studies, United Kingdom.

LeBlanc, D., R. Goodsell, Z. Bareket, M. R. Hagan, M. L. Buonarosa, J. Devonshire, S. E. Bogard, R. D. Ervin, C. B. Winkler, J. R. Sayer. 2006. *Road Departure Crash Warning System Field Operational Test: Methodology and Results, Volume 1: Technical Report.* Technical Report UMTRI-2006-9-1. Ann Arbor, MI: University of Michigan Transportation Research Institute.

Lee, J., J. Forlizzi, and S. E. Hudson. 2008. Iterative design of MOVE: A situationally appropriate vehicle navigation system. *Int J Hum Comput Stud* 66:198–215.

Mahajan, V., E. Muller, and F. M. Bass. 1990. New product diffusion model in Marketing: A review and directions in research. *J Mark* 54:1–26.

Manes, D., P. Green, and D. Hunter. 1998. *Prediction of Destination Entry and Retrieval Times Using Keystroke-Level Models.* Technical Report UMTRI-96-37. Ann Arbor, MI: University of Michigan, Transportation Research Institute.

Mayhew, D. R., H. M. Simpson, and A. Pak. 2003. Changes in collision rates among novice drivers during the first months of driving. *Accid Anal Prev* 35(5):683–91.

McCartt, A. T., L. A. Hellinga, and K. A. Bratiman. 2006. Cell phones and driving: Review of research. *Traffic Injury Prevention* 7:89–106.

McEvoy, S. P., M. R. Stevenson, A. T. McCartt, M. Woodward, C. Haworth, P. Palamara, et al. 2005. Role of mobile phones in motor vehicle crashes resulting in hospital attendance: A case-crossover study. *Br Med J* 1–5. http://bmj.bmjjournals.com/cgi/content/abstract/bmj.38537.397512.55v1 (accessed April 3, 2007).

McEvoy, S. P., M. R. Stevenson, and M. Woodward. 2007. The contribution of passengers versus mobile phone use to motor vehicle crashes resulting in hospital attendance by the driver. *Accid Anal Prev* 39:1170–6.

Michon, J. A., ed. 1993. *Generic Intelligent Driver Support*. London: Taylor & Francis.

Nowakowski, C., and P. Green. 2000. *Prediction of Menu Selection Times Parked and While Driving Using the SAE J2365 Method.* Technical Report 2000-49. Ann Arbor, MI, University of Michigan, Transportation Research Institute.

Nowakowski, C., D. Friedman, and P. Green. 2001. *Cell phone ring suppression and HUD caller ID: Effectiveness in reducing momentary driver distraction under varying workload levels*. Technical report 2001-29. Ann Arbor, MI: University of Michigan, Transportation Research Institute. http://www.umich.edu/,driving/publications.html(accessedApril3,2007).

Nowakowski, C., Y. Utsui, and P. Green. 2000. *Navigation System Evaluation: The Effects of Driver Workload and Input Devices on Destination Entry Time and Driving Performance and their Implications to the SAE Recommended Practice*. Technical Report UMTRI-2000-20. Ann Arbor, MI, University of Michigan, Transportation Research Institute.

Olson, J. R., and E. Nilsen. 1987–1988. Analysis of the cognition involved in spreadsheet software interaction. *Hum Comput Interact* 3:309–49.

Owens, J. M., S. B. McLaughlin, and J. Sudweeks. 2010. *On-Road Comparison of Driving Performance Measures When Using Handheld and Voice-Control Interfaces for Mobile Phones and Portable Music Players*. SAE paper 2010-01-1036. Warrendale, PA: Society of Automotive Engineers.

Peacock, B., and W. Karwowski. 1993. *Automotive Ergonomics*. London: Taylor & Francis.

Pettitt, M. 2008. Visual demand evaluation methods for in-vehicle interfaces. Unpublished PhD dissertation. Nottingham, England: University of Nottingham.

Pettitt, M., G. Burnette, and A. Stevens. 2007. An extended keystroke level model (KLM) for predicting the visual demand of in-vehicle information systems. In *CHI 2007 Proceedings*, 1515–24. San Jose, CA, New York: ACM.

Piechulla, W., C. Mayser, H. Gehrke, and W. König. 2003. Reducing Drivers' mental workload by means of an adaptive man-machine interface. *Transp Res Part F Traffic Psychol Behav* 6:233–48.

Ranney, T. A. 2008. *Driver Distraction: A Review of the Current State-of-Knowledge.* Technical Report DOT HS 810 787. Washington, DC: U.S. Department of Transportation, National Highway Traffic Safety Administration.

Redelmeier, D. A., and R. J. Tibshirani. 1997. Association between cellular-telephone calls and motor vehicle collisions. *N Engl J Med* 336:453–8.

Redelmeier, D. A., and R. J. Tibshirani. 2001. Car phones and car crashes: some popular misconceptions. *Can Med Assoc J* 164(11):1581–2.

Regan, M. A., J. D. Lee, and K. Young. eds. 2008. *Driver Distraction: Theory, Effects, and Mitigation*. Boca Raton, FL: CRC Press.

Ribbens, W. B., and D. E. Cole. 1989. *University of Michigan Automotive electronics Delphi*. Ann Arbor, MI: Office for the Study of Automotive Transportation, University of Michigan, Transportation Research Institute.

Richardson, B., and P. Green. 2000. *Trends in North American Intelligent Transportation Systems: A year 2000 Appraisal.* Technical Report UMTRI-2000-9. Ann Arbor, MI, University of Michigan, Transportation Research Institute.

Roskam, A. J., K. A. Brookhuis, D. deWaard, O. M. J. Carsten, L. Read, S. Jamson, et al. 2002. *Development of Experimental Protocol.* HASTE Deliverable 1. Brussels, Belgium: European Commission.

Ross, T., K. Midtland, M. Fuchs, A. Pauzie, A. Engert, B. Duncan, et al. 1996. *HARDIE Design Guidelines Handbook: Human Factors Guidelines for Information Presentation by ATT Systems*. Luxembourg: Commission of the European Communities.

Salvucci, D. D. 2009. Rapid prototyping and evaluation of in-vehicle interfaces. *ACM Trans Comput Hum Interact* 16(2):9:1–33.

Salvucci, D. D., D. Markley, M. Zuber, D. P. Brumby. 2007. iPod distraction: Effects of portable music-player use on driver performance. In *CHI Proceedings*, San Jose, California. New York: ACM.

Salvucci, D. D., M. Zuber, E. Beregovaia, and D. Markley. 2005. Distract-R: Rapid prototyping and evaluation of in-vehicle interaces. In *CHI 2005 Proceedings*. New York: Association for Computing Machinery.

Schindhelm, R., C. Gelau, A. Keinath, K. Bengler, H. Kussmann, P. Kompfner, et al. 2004. *Report on the Review of Available Guidelines and Standards*. AIDE deliverable 4.3.1. Brussels, Belgium: European Commission. http://www.aide-eu.org/pdf/sp4_deliv/aide_d4-3-1.pdf (accessed April 4, 2007).

Schweitzer, J., and P. A. Green. 2007. *Task Acceptability and Workload of Driving Urban Roads, Highways, and Expressways: Ratings from Video Clips*. Technical Report UMTRI-2006-6. Ann Arbor, MI: University of Michigan Transportation Research Institute.

Shutko, J., K. Mayer, E. Laansoo, and L. Tijerina. 2009. *Driver Workload Effects of Cell Phone, Music Player, and Text Messaging Tasks with the Sync-Voice Interface versus the Devices' Handheld Visual-Manual Interfaces*. SAE paper 2009-01-0768, Warrendale, PA: Society of Automotive Engineers.

Sloss, D., and P. Green. 2000. *National Automotive Center 21st Century Truck (21T) Dual Use Safety Focus*. SAE Paper 2000-01-3426. Warrendale, PA: Society of Automotive Engineers (published in National Automotive Center Technical Review, Warren, MI, U.S. Army Tank-Automotive and Armaments Command, National Automotive Center, 63–70).

Society of Automotive Engineers. 2002. Calculation of the Time to Complete In-Vehicle Navigation and Route Guidance Tasks. SAE recommended practice J2365. Warrendale, PA: Society of Automotive Engineers.

Society of Automotive Engineers. 2004a. *Navigation and Route Guidance Function Accessibility While Driving.* SAE recommended practice 2364. Warrendale, PA: Society of Automotive Engineers.

Society of Automotive Engineers. 2004b. *Rationale Document for SAE J2364.* SAE information report J2678. Warrendale, PA: Society of Automotive Engineers.

Society of Automotive Engineers. 2010a. *Definitions of Driving Performance Measures and Statistics.* SAE recommended practice J2409, draft. Warrendale, PA: Society of Automotive Engineers.

Society of Automotive Engineers. 2010b. *SAE Handbook 2010*. Warrendale, PA: Society of Automotive Engineers.

Steinfeld, A., and H.-S. Tan. 2000. Development of a driver assist interface for snowplows using iterative design. *Transp Hum Factors* 2(3):247–64.

Steinfeld, A., D. Manes, P. Green, and D. Hunter. 1996. *Destination Entry and Retrieval with the Ali-Scout Navigation System.* Technical Report UMTRI-96-30, also released as EECS-ITS LAB FT97-077. Ann Arbor, MI: University of Michigan, Transportation Research Institute.

Stevens, A., A. Quimby, A. Board, T. Kersloot, and P. Burns. 2004. *Design Guidelines for Safety of In-Vehicle Information Systems*. Crowthorne, UK: TRL Limited.

Stutts, J. C., D. W. Reinfurt, L. Staplin, and E. A. Rodgman. 2001. *The Role of Driver Distraction in Traffic Crashes*. Washington, DC: AAA Foundation for Traffic Safety. http://www.aaafts.org/pdf/distraction.pdf (accessed April 3, 2007).

Teo, L., and B. E. John. 2008. Towards predicting user interaction with CogTool-Explorer. In *Proceedings of the Human Factors and Ergonomics Society 52nd Annual Meeting*, New York, Sept 22–26, 2008. Santa Monica, CA: Human Factors and Ergonomics Society.

Tijerina, L., L. Angell, A. Austria, A. Tan, and D. Kochhar. 2003. *Driver Workload Metrics Literature Review*. Washington, DC, U.S. Department of Transportation, National Highway Traffic Safety Administration.

Tsimhoni, O., D. Smith, and P. Green. 2004. Address entry while driving: Speech recognition versus a touch-screen keyboard. *Hum Factors* 46(6):600–10.

U.S. Department of Transportation. 2008. *National Motor Vehicle Crash Causation Survey.* Technical Report DOT HS 811 059. Washington, DC: U.S. Department of Transportation, National Highway Traffic Safety Administration.

U.S. Department of Transportation. 2009a. 2008 *Traffic Safety Annual Assessment—Highlights.* Technical Report DOT HS 811 172. Washington, DC: U.S. Department of Transportation. www-nrd.nhtsa.dot.gov/pubs/811172.pdf (accessed May 30, 2010).

U.S. Department of Transportation. 2009b. *Driver Electronic Device Use in 2008.* Technical Report DOT HS 811 184. Washington, DC: U.S. Department of Transportation. www-nrd.nhtsa.dot.gov/Pubs/811184.pdf (accessed April 3, 2007).

Underwood, S. E. 1989. *Summary of Preliminary Results from a Delphi Survey on Intelligent Vehicle-Highway Systems*. Technical Report. Ann Arbor, MI: University of Michigan.

Underwood, S. E. 1992. *Delphi Forecast and Analysis of Intelligent Vehicle-Highway Systems Through 1991: Delphi II.* Ann Arbor, Program in Intelligent Vehicle-Highway Systems. IVHS Technical Report-92-17. Ann Arbor, MI: University of Michigan.

Underwood, S. E., D. Chen, and R. D. Ervin. 1991. Future of intelligent vehicle-highway systems: A Delphi forecast of markets and socio-technological determinants. *Transp Res Rec* No. (1305):291–304.

Violanti, J. M., and J. R. Marshall. 1996. Cellular phones and traffic accidents: An epidemiological approach. *Accid Anal Prev* 28:265–70.

Wang, J. S., R. R. Knipling, and M. J. Goodman. 1996. The role of driver inattention in crashes: New statistics from the 1995 crashworthiness data system. In *Association for the Advancement of Automotive Medicine 40th Annual Conference Proceedings,* 377–92. Des Plaines, IL: Association for the Advancement of Automotive Medicine.

Wards Automotive Group. 2009. *World Motor Vehicle Data 2009*. Southfield, MI: Wards Automotive Group.

Wejnert, B. 2002. Integrating models of diffusion of innovations: A conceptual framework. *Annu Rev Sociol (Annu Rev)* 28:297–306.

Wierwille, W. 1995. Development of an initial model relating driver in-vehicle visual demands to accident rate. In *Proceedings of the Third Annual Mid-Atlantic Human Factors Conference,* 1–7. Blacksburg, VA: Virginia Polytechnic Institute and State University.

World Health Organization. 2009. *Global Status Report on Road Safety*. Geneva, Switzerland: World Health Organization.

Young, R., B. Aryal, M. Muresan, Z. Ding, S. Oja, and N. Simpson. 2005. Road-to-Lab: Validation of the static load test for predicting on-road driving performance while using advanced in-vehicle information and communication devices. In *Driving Assessment 2005: Proceedings of the Third International Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, 240–54. Iowa City, IA: University of Iowa. http://ppc.uiowa.edu/driving-assessment/2005/final/index.htm (accessed April 3, 2007).

Young, K., M. Regan, and M. Hammer. 2003. *Driver Distraction: A Review of the Literature.* Report 206. Melbourne, VIC Australia: Monash University.

33 Human–Computer Interaction in Aerospace

Steven J. Landry

CONTENTS

33.1 HUMAN–COMPUTER INTERACTION IN AEROSPACE

Aviation's interest with concepts relevant to human–computer interaction (HCI) began with the need to understand how pilots could interact with mechanical displays that indicated such things as airspeed, altitude, aircraft orientation, heading, and bearing from a radio navigation aid. When aircraft technology progressed and displays became more numerous and more complex, the interface issues became more pronounced. As travel by aircraft was commercialized, the consequences of these interface issues became more severe. When modern air traffic control was introduced, a new arena for human interface issues was opened. By the 1970s, computers were in commercial aircraft, and air traffic control was utilizing radar displays and automated flight data processing. This progression has not stopped, as higher levels of automation and new types of interfaces have been steadily added to flight decks and air traffic control rooms over the succeeding decades.

This continuous progression of displays and automation has taxed the capabilities of researchers to fully investigate the vast and varied challenges associated with aviation HCI. This chapter discusses the efforts that have been made, are being made, and also those issues which are just now on the horizon.

33.1.1 Basic Challenges, Solutions, and Principles

Human-factors issues related to aviation displays were recognized as early as 1923, as indicated by the following quotation from a National Advisory Committee for Aeronautics (the predecessor to NASA) report:

> The reaction of the aviator to his instruments has to be considered, as well as the operation of the instruments themselves. This is evident enough in the case of appliances such as oxygen apparatus, intended solely for the comfort and efficiency of the aviator, or in the case of complicated instruments such as bomb sights. But it is equally true with the more simple, direct-reading instruments. It is not enough for such instruments to be mechanically correct; they must be, in the case of service instruments, readily intelligible to the pilot. The manipulation of the instrument must not make an appreciable demand on his time or attention. The visibility must be satisfactory both day and night. Finally, service instruments must be "fool-proof." Although much can be accomplished by technical instruction courses for aviation personnel, still the personal prejudice of the average pilot has to be reckoned with. If the instrument for any reason fails to appeal to the individual pilot, he will take great chances rather than trouble to look at it. On the other hand, if the instrument pleases his fancy, he may grow so attached to it that he will claim he could not fly safely without it, even though the instrument is scientifically known to be incorrect. Curious examples of this circumstance were found in the popularity of the earlier liquid type Pitot tube among the British pilots and the spinning-top inclinometer among the French (Hersey 1923, pp. 8–9).

Another early example of aviation cognitive engineering research was in 1939, when an Applied Psychology Unit at Cambridge University in England, headed by Frederic Bartlett, began studying the design of aviation equipment (Bartlett 1943). Bartlett's group studied manual control performance, the interrelationship between controls and displays, and human vigilance. Also in 1939, the National Research Council in the United States created a Council on Aviation Psychology. The U.S. Army and Navy created programs on aviation psychology in 1940 and 1945, respectively.

World War II provided a wealth of aviation experience, and, after the war was over, aviation-savvy graduate students entering universities on the GI Bill. At the same time, advances were being made in navigation systems technology, cathode ray tube (CRT) displays, aircraft instrumentation, and control technology. The rapid increase in civil aviation made the development of air traffic control a priority as well, and the advent of radar displays peaked interest in cognitive engineering issues for air traffic controllers.

These developments led to an explosion of research in the 1950s, both in the United States and in Europe. One of the more influential reports was a 1951 blue ribbon committee report titled "Human Engineering for an Effective Air-Navigation and Traffic-Control System" (Fitts 1951b). This paper summarized the state of research as it concerned air transportation and identified key areas of future research. As such, it provides a convenient milepost for a discussion of the history and progress of aviation cognitive engineering research.

This chapter will begin from a historical perspective, covering the major issues conceived of in the 1950s and how they were addressed in the succeeding decades, if at all. Following that will be a discussion of modern concerns, which grew out of the computerization of flight decks and innovative display concepts such as heads-up displays (HUDs). Lastly will be a discussion of the future challenges for aviation and air traffic control displays.

One concern for researchers in the 1950s regarded the display of information on "transitory displays," mainly dials and gauges. As such the main focus was on whether (and how) to integrate information on a single display, how to arrange displays, how displays, controls, and the comprehension of spatial information were related, and even what types of displays were best for certain functions.

Many of these questions have been successfully addressed and provide today's engineers with some of the most concrete guidance that cognitive engineering has to offer. Some of them continue to be fascinating topics of research to this day, and, of course, many new topics related to information display and assessment have arisen over the years.

33.1.1.1 Single-Sensor Single-Instrument Displays versus Integrated Displays

The problems inherent in single-sensor single-instrument (SSSI) displays have become increasingly obvious as the number and complexity of sensors and other information sources has increased. This has been particularly noticeable in aviation, where flight decks have undergone significant changes since 1951, but has also been true for nuclear plants, which were just beginning to be tested in 1951. Interestingly, air traffic controllers have only recently begun to encounter this problem as the number of displays, and elements on those displays, provided to a controller and in the Traffic Management Unit has increased substantially.

SSSI displays have been gradually replaced or incorporated into multifunction displays as display technology has improved, although there are still examples of traditional SSSI displays on older aircraft and plants. (It is an expensive venture to replace them with graphical displays.) It seems clear from the research however, that fewer displays are better than many displays, although the former is not without its own problems.

The reason that SSSI displays are generally considered to be worse than an integrated display seems, in part, to be related to the fact that a person's ability to monitor dynamic information is very low, even when only keeping track of a few items (Monty 1973). Also, operators of aircraft have demonstrated that they have difficulty in integrating a large amount of displayed information from different sources (Murphy et al. 1978). These findings mean that it is difficult for an operator to maintain comprehension of a dynamic situation when forced to scan from display to display.

A study of the operation of manipulators showed subjects performed 30%–40% better using two displays as opposed to four displays, despite a greater amount of information in the four-display case (Bejczy and Paine 1977). This result extended a much earlier finding showed that there would be fewer missed detections if there were, for example, five times the error rate on one display compared with five displays each with one-fifth the error rate (Conrad 1951).

The presence of large numbers of displays also increases the amount of irrelevant data, also called "clutter," through which the operator must sort. Almost by definition, SSSI displays must, at every moment, display all the information that may be needed in any state of the system. Because of this, there must be irrelevant information present at any given time, and the operator must filter out this irrelevant information. As the quantity of irrelevant information increases, operator performance decreases, particularly if the information is similar (Dorris et al. 1977; Hodge and Reid 1971; Well 1971).

In addition, relevant data may be spread out over different, and often physically separated, instruments. In such a case, the information may need to be integrated mentally to obtain a higher-order representation. This operation can be cognitively intensive, and during situations of stress or short time constraints, it may not be possible (Vicente and Rasmussen 1992).

In the next section, these issues—general design principles, arrangement of displays, and clutter—will be discussed in more detail. Many of the same or similar principles can be noticed in Chapter 14 of this volume.

33.1.1.2 Basic Display Principles

In examining the question of whether fewer displays containing integrated information were better, and how information may be integrated, researchers have developed a number of principles for the design of displays. In particular, researchers have studied problems related to discerning the direction of motion of a pointer, and in the use of multipointer displays, which have been used in altimeter designs (Fitts 1951a). Since then, a number of design principles with regard to such quantitative displays have been put forth (Sanders and McCormick 1993).

One study compared digital speedometers with dials and curvilinear displays, all of which were generated electronically. For accuracy and speed of reading, the digital display performed consistently better (Simmonds, Galer, and Baines 1981). In general, digital displays have been found superior when a precise numeric value is required, and when the values presented remain visible long enough to be read.

Fixed-scale displays with moving pointers are better in the opposite cases—where precision is not required or the values are changing quickly. They are also better in cases where direction and rate of change of the values are important. For the design of analog displays, the following guidelines have been generally accepted (Heglin 1973):

- A moving pointer against a fixed scale is preferred.
- Small changes in quantity are more visible with a moving pointer display.
- If the quantity to be read is analogous to some physical interpretation, horizontal, or vertical fixed scales with moving pointers should be used to provide a zero reference.
- Do not mix multiple moving-element indicators when they are related to the same function, as this can cause reversal errors.
- Moving scale displays can be used where the scale is too great to be displayed on the face of an instrument.
- If a control changes the quantity on the display, it is less ambiguous for the pointer to move than the scale.

In addition, for quantitative displays, the numeric progression and length of scale units affect the speed and accuracy of quantitative reading (Whitehurst 1982), and scale progressions should be in increments of 1s or 5s (to ease determination of readings in multiples of 1 and 5). Unusual progressions and decimals should be avoided. It was also found that the distance between graduation markers should not be less than 0.05–0.07 inches apart (greater under low-illumination conditions), that scale markers should be used, and they should have graduation marks for the lowest unit to be read, but in no case lower than one-fifth or one-tenth (Cohen and Follert 1970). Bar-type displays should only show one segment of the bar, should not have the bar extended to zero, and should not be used where a need for an indication at zero is needed (Green 1984).

Altimeter designs were of particular concern in the late 1940s due to a spate of accidents related to misreading them. One of the early altimeters responsible for some of the problems violated one of the principles above: it had three pointers on the same scale. The operator would have to read three pieces of information and integrate them to get altitude. In 1968, a study determined that an integrated vertical display of altitude was best in terms of time and accuracy of reading (Roscoe 1968). In explaining the results, the researcher indicated that key factors were the analogy of the vertical instrument to altitude and the integration of all elements of altitude into one display.

Computers have changed how instruments are driven; instead of mechanical devices, sensors provide digital input to computers, which drive displays. However, despite the nature of the digital computers driving them, modern graphical displays often show representations of analog-like displays for instruments. This is because analog displays were found to be more quickly read for this purpose, and contributed less to workload, compared with digital displays (Hanson et al. 1981).

In addition to quantitative displays, aviation instruments may be used for check reading or qualitative purposes (Sanders and McCormick 1993). These instruments, and some principles for their design, are discussed in Sections 33.1.1.2.1 and 33.1.1.2.2.

33.1.1.2.1 Check-reading Display Research

Researchers demonstrated that for a number of instruments configured together for check reading, the pointers on round instruments were best positioned at 9 or 12 o'clock (Sanders and McCormick 1993). When all pointers are showing nearly identical readings, the grouping of the instruments and positioning of the pointers appeal to the similarity gestalt of human cognition. Violation of the gestalt is immediately and reliably recognized. Further research indicated that the addition of a line joining the instruments at their null position improved or added to the gestalt and improved performance (Dashevsky 1964).

33.1.1.2.2 Qualitative Instrument Research

The distinction does not appear to have been made between quantitative and qualitative, perhaps a subset of which is check reading, instruments before 1960. Qualitative instruments are distinct from quantitative in that the specific numerical value is irrelevant; instead the approximate value, its value in relation to some standard, its rate, or its trend are of interest. The difference may have first been noticed in a study that examined two types of readings of an instrument using three types of scales. In that study, subjects had to either read the quantitative value, or read high, low, or OK. Open window displays (see Figure 33.1) were read the quickest for quantitative values, but vertical scales were read quickest for the qualitative assessment (Elkin 1959).

Qualitative scales also typically have some identification of ranges. This can be done through the use of color, or by the addition of coded markings that have some association to the meaning of the range (Sabeh, Jorve, and Vanderplas 1958).

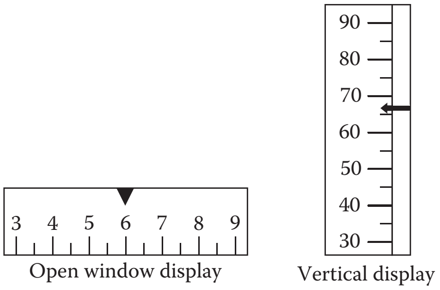


FIGURE 33.1 Display types.

33.1.1.3 Arrangement of Displays

A second question brought up in the Fitts report (1951b) regarded the organization of the display elements. A number of advances in relation to this arrangement problem were made shortly after its publication. Extensive studies, most notably one in which Fitts himself participated (Fitts, Jones, and Milton 1950), were underway examining visual scanning patterns to determine the optimal layout of instruments. These studies led the Civil Aeronautics Board (later to become the FAA) to establish a standard arrangement for flight instruments, commonly referred to as the "basic T," in 1953 (Civil Aeronautics Board 1953). It included six basic instruments, namely airspeed, attitude, altitude, direction, "climb," which is taken here to mean vertical speed, flight path deviation, and their required arrangement. This was later reduced to just the first four instruments (Civil Aeronautics Board 1957).

The analysis used in relating visual scanning patterns to display positioning was referred to as "link analysis." In this analysis, a "link" was sequential fixations on two items. The more occurrences of sequential fixations there are between two items, the stronger the link is considered to be. Items strongly linked should then be located next to one another, a technique and principle used in other fields, such as in operation centers onboard naval vessels (Chapanis 1959). Other studies of eye movements determined that the best location for important information is just above or below the glareshield of a vehicle (Cole, Milton, and McIntosh 1954). Since then, advances have been made in automating information organization (Mendel and Sheridan 1986) and instrument panel layout (Pulat and Ayoub 1979), although methods of this sort are not widely used.

The layout of the instruments not only serves to reduce the time to scan the instruments, but can also affect workload. A principle of display design called the "proximity compatibility principle" states that, when attempting to integrate two sources of information from two locations, greater workload is induced by greater separation, as the physical proximity of the sources should mimic the cognitive proximity of the information (Wickens and Carswell 1995).

Multifunction graphical displays (MFDs), common in modern aircraft, have remained consistent with the basic T and other principles, despite the substantial difference in technology, although notable differences exist between MFDs and SSSI displays. MFDs provide, in one viewport, a number of elements that may need to be integrated frequently, satisfying the proximity compatibility principle (Seidler and Wickens 1992). The content of the MFD can be manually controlled, providing the opportunity to present a wealth of information for which there was no physical space previously.

However, two problems arise. First, information that is not currently displayed is unavailable to the pilot. If that information changes, then the pilot will not be aware of it. Second, the "depth" added to the display presents the problem of navigation, although this can be mitigated through good organization and minimal depth of the pages (Allen 1983; Francis 2000; Schneiderman 1998). However, if these principles are not met, there is the possibility that the operator will become lost in the database. In addition, there is workload and heads-down time associated with the controls for the display.

33.1.1.4 How Much Information Can Be Displayed Effectively?

The problem of how much information can be fit on a display has significantly evolved since 1951, but is still a topic of debate and research. A large body of work on visual processing has been amassed over the years, and many of the theories are still hotly debated. What is certain is that the distinguishability of items on a display is overwhelmed by the somewhat limited perceptual processing capacity of the human. Without going into a great amount of detail, some of the findings with respect to this issue will be summarized.

There are two fundamental limits on extracting information from a visual scene: processing and attention. At any time, only a small amount of the information available to the retina can be processed (although all of it is perceived). The mechanics of processing are still being determined, but it appears likely that some high-level representation is quickly obtained, followed by a "weeding out" of some portions of the image (VanRullen and Thorpe 2001). Scene processing allows a variety of information to be extracted, but specific objects of interest may or may not be extracted in sufficient detail. Focused attention captures this detail, but objects can only be processed serially, and subsequent processing of these objects is inhibited (Tipper and Driver 1988). (This effect is referred to as "negative priming.") The combination of these effects yields an effective capacity of four to six items (VanRullen and Koch 2003). This number represents how many items can be represented in visual awareness at any one time. Objects/information may then be transferred to long-term memory, a process that has also been researched extensively.

How much information can be fit on one display, then, is dependent on the task and environmental factors. If one is reading a depiction of an instrument approach with the intention of memorizing details, and there is no time pressure, a great deal of information can be displayed and recalled effectively. If an air traffic controller has a fraction of a second to glean track information from a radar display, he or she is unlikely to extract more than four to six pieces of information from that display. Moreover, what information would be extracted from that display is unclear, although factors of the displayed information, such as salience, size, and color, can influence selective attention.

One further question remains with regard to information display that we will address here—the rate at which information can be extracted. This also is complex, dependent on physiological factors, features of the display, complexity of the information to be extracted, and other factors. Simple responses to visual stimuli can occur within 30–60 milliseconds (Iwasaki 1996), but more complex objects such as text can require a great deal more time.

In summary, the answer to the question of how much information can be fit on a display is—it depends. The question has become even more relevant as electromechanical displays have been almost universally replaced with electronic displays driven by computers, which can display a great deal of information in one place. These displays are also three dimensional (3D) in the sense that they can contain pages of information, which adds a new task—navigation of a display and their contents can be modified by the user or automation.

33.1.1.5 Clutter

Clutter is a significant issue for multifunction displays, such as modern air traffic control and flight deck displays that show combined depictions of route, weather, and traffic, as well as for HUDs. The source of the clutter is different in the two cases. In the former case, a large quantity of data is being overlaid on the display, whereas in the latter the display is overlaid on a source of great clutter—the outside world. In both cases, the problem is the same: response time for a stimulus is increased with the proximity of nonstimulus items (Eriksen and Eriksen 1974). Additionally, clutter can increase the time it takes for the pilot to locate an item, or even obscure items of interest (Wickens 2003). Clutter has also been mentioned by pilots in subjective ratings for some time (Abbott et al. 1980).

The solution to this problem is to provide the operator with decluttering capability—that is, the ability to remove information from the display, either manually or automatically. However, if the information is no longer present, it cannot be integrated with other information, and changes to the information, which could be important, will of course not be noticed. This is in addition to the workload required to remove or add information to the display, which could be considerable, particularly in the case of navigating a multifunction display.

33.1.1.6 Displaying Spatial Relationships

The Fitts report (1951b, p. xvii) suggested that: "studies should be made to determine principles governing the effective display of information about the relative position of aircraft and ground objects in tri-dimensional space." Specifically, the committee felt that research should be directed at determining the best types of projections for displaying spatial relations, and how symbolic and pictorial displays are best used for this purpose. In addition, more studies were called for to determine what display characteristics were required when control manipulation resulted in changes in the display. For example, turning a control yoke results in changes in bank and heading indications.

Aviation displays is a natural place for interest into the display of spatial relations to develop. Air traffic controllers must gauge both absolute and relative position and motion of aircraft in at least two dimensions (often three), and provide spatial control guidance to pilots. A pilot must, through use of instrumentation, determine relative and sometimes absolute position in 3D coordinates, and keep track of 3D orientation. This is accomplished through the use of an attitude indicator (ADI), which shows pitch and bank; a heading indicator, which shows absolute orientation; a turn indicator, which shows yaw rate; and one or more instruments showing relative position and bearing (e.g., altimeters, radio magnetic indicator, course deviation indicator, vertical deviation indicator, distance measuring equipment).

One problem which almost certainly drove the recommendation in the Fitts report (1951b) regarding the display of orientation was the problem of the ADI. In particular, what should move—the aircraft or the earth? Several principles related to this question have been put forth in consideration of the body of research into interfaces (Roscoe, Corl, and Jensen 1981) as follows:

- Principle of pictorial realism: The display should be pictorially analogous to the real world.
- Principle of compatible motion: The part that moves in the real world should be the part that moves on the display.
- Principle of integration: Related information should be integrated on the same display.
- Principle of pursuit presentation: A display should allow pursuit tracking versus just compensatory tracking; that is, if a target moves both absolutely and relatively, both sources of movement should be displayed.

A moving aircraft on an ADI or an air traffic controller's radar scope would be compatible with the principle of compatible motion. However, while the radar scope also complies with the principle of pictorial realism, from the pilot's perspective the moving aircraft would (loosely) violate it—it appears from the pilot's perspective that the horizon is banked, that is, the earth moves, as opposed to the aircraft. Although designers have settled on the moving earth model for ADI design, the debate was never satisfactorily resolved. Experiments based on a proposal to integrate the two models (Fogel 1959) showed improved performance (Beringer, Willeges, and Roscoe 1975). Researchers have argued that the moving earth ADI has continued to be used only because of the lack of interest on the part of pilots to change it, despite evidence that it is has been causing accidents for decades (Bryan, Stonecipher, and Aron 1954; Roscoe 1997).

In addition to the orientation question addressed mainly by the ADI, there are questions about how best to portray relative position. This is a common problem for both air traffic controllers, who must assess relative bearing and range between aircraft, and pilots, who must frequently assess their relative bearing and range from some navigational fix. More recently, the question of vertical navigation has received attention.

A typical horizontal navigation problem for a pilot is common to anyone familiar with basic instrument flying. From any given position, a pilot could be asked to fly to a "fix," whose position is defined by bearing and range relative to a radio navigation aid (e.g., a very high frequency omni-directional radio range, or VOR). Although this can be calculated from current heading and the ownship's bearing and range to the VOR, it is not a trivial task. A simpler task is to fly inbound on a course to a VOR or outbound on a radial from a VOR. Another example is the task of determining the position of the ownship given access to several radio navigational aids.

Recent research has qualified the difficulties inherent in these operations as "representational" problems (Zhang and Norman 1994). The theory follows from the argument that the information to accomplish many tasks is distributed between the external stimuli (the interface) and internal cognition (in this case of the pilot) (Norman 1993). Furthermore, the information representations may have different scales: ratio, interval, ordinal, or nominal. Efficient external representations reflect all the categories of information present in the "real-world" object. Deficiencies in external representations would require that information be distributed to the internal cognition of the operator, whereas surplus information, here defined as information present in the display that is not in the object, could be misleading.

Application of this theory to the problem of horizontal navigation shows that different equipment requires different levels of internal cognition. A simple automatic direction finder (ADF) or VOR system requires internal computation of angular differences and range. However, a map type display explicitly represents these, simplifying the task.

An extreme example of this representational problem is the difficulty in determining relative position from terrain or weather on a paper map (or even presented aurally!) from their knowledge of absolute position as given by the flight instruments. It is difficult and time-consuming to accurately comprehend this situation. This was apparently recognized, resulting in the display of "minimum safe" altitudes, which changes the nature of what needs to be comprehended (i.e., the pilot only must know his or her position with respect to that altitude, a one-dimensional, relative task).

33.1.1.7 Situation Awareness

It is a notable theme that modern research attention has turned from information display, as described above, to information assessment. As complex automation has been added to flight decks and air traffic facilities, and as more information has been provided to operators, accessing and comprehending that automation and information has become problematic. Researchers began to understand that not only did they need to know how to display information, but also how operators represented and utilized that information, to provide a better match between technology and the human.

Interest in information assessment has lead to a significant body of research into situation awareness (SA). This has been a particularly important concept in aviation, because safe and successful operation in aviation requires a great deal of knowledge about the environment outside an individual aircraft. Many aviation accidents can be attributed to a lack of knowledge of the environment external to the aircraft (e.g., collisions with aircraft and the ground). For air traffic controllers, their knowledge of the "big picture" includes not only the states of the aircraft in their own sector, but those of neighboring sectors, airports, weather, and much more. Attempts to understand how pilots and controllers obtain and maintain this "big picture" have been the focus of SA research.

Military aviators, since as far back as the First World War, have understood the importance of SA, and instructed their pilots in developing and maintaining good SA. One of Germany's top World War I fighter aces, Oswald Boelcke, listed among his "dicta" that "the pilot must acquire the habit of 'taking in' unconsciously the general progress of the whole multiaircraft dogfight going on around the individual combat in which the pilot will become involved … (so that) no time (is) wasted in assessment of the general situation after the end of an individual combat" (Hacker 1984). Boelcke also prescribes knowledge of one's own machine, the enemy's machine, and navigational fixes. SA has been a significant part of aeronautical training since that time.

As a research topic, however, the concept was mostly ignored by researchers until the 1980s. Due to increasing flight deck and air traffic automation, pilots' and air traffic controllers' role as supervisor of these systems was increasing, reducing the time they could spend in developing SA. At the same time, a great deal of new sensor information was becoming available to designers of aviation automation, information that could be used to reinforce the controllers' and pilots' SA. Researchers began looking into what SA is, what affects an operator's ability to construct SA and to keep it, and how it might be measured.

Many definitions of SA exist, with most agreeing that SA is, at least in part, the comprehension of elements of the environment that have (or may have) some bearing on the task being accomplished. Some efforts have viewed SA as a static, information-driven product; some have viewed it as a dynamic process, whereas others have viewed SA as a high-level description of certain aspects of task behavior. A significant body of work has also gone toward determining how to measure SA.

One of the most widely quoted definitions of SA is: "… the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" (Endsley 1988, p. 97). Another researcher has called SA "… an integrated understanding of factors that will contribute to the safe flying of the aircraft under normal or non-normal conditions" (Regal, Rogers, and Boucek 1988, p. 65). These definitions attempt to describe SA as a product, something that an operator either has or does not have. As such, errors resulting from a lack of SA can be studied. Researchers have found that controllers, viewing recorded air traffic files, were unable to recall many details about traffic when asked about them during the scenario, including call signs, control level (who has control of the aircraft), altitude, speed, heading, and often even failed to report that some aircraft were present at all (Endsley and Rodgers 1996). Other researchers studying error report databases have found that in only about 20% of serious operational air traffic errors were controllers aware of the problem developing (Gosling 2002).

Researchers who view SA as a product often defer the process of obtaining and maintaining SA to a separate process, often referring to it as "situation assessment." This type of SA has been described as "(an) adaptive, externally-directed consciousness" (Smith and Hancock 1995, p. 137) and "the integration of knowledge resulting from recurrent situation assessments" (Sarter and Woods 1991, p. 45). Here, the sources of information important to SA (visual, auditory, tactile, other sensory input, knowledge of procedures and regulations, etc.) are diverse and often clearly identifiable. The operator's ability to obtain SA begins with the most fundamental cognitive processes (detection), and progress to very sophisticated concepts (comprehension and projection).

Some researchers have called into question the utility of the SA concept. Many feel it is a high-level concept that does not have sufficient granularity to really explain anything. SA has been referred to as a description of observations of humans operating complex systems in a dynamic environment (Billings 1995) and as simply equivalent to expertise (Crane 1992). As such, SA is a description of a set of cognitive processes that are used together, and is really only useful in categorizing or grouping behaviors and errors. Some researchers feel that considering SA as a causal agent can be counterproductive to understanding operator behavior (Flach 1995).

No one set of measures has been clearly identified for SA. Typical measures can be broken into three categories: (1) explicit (or knowledge-based), (2) implicit (or performance-based), and (3) subjective (Vidulich 1992). Examples of explicit methods include participants' recall of a situation (post hoc), an ongoing narrative provided concurrently with the task, or freezing a scenario and questioning the participant about the decisions, events, or the task environment. These measures can be compared with actual state of the system to provide a better measure, but have been criticized as too subjective (in the case of recall) (Fracker 1991), or too intrusive (in the case of freezing a scenario) (Sarter and Woods 1991).

Implicit measures examine task performance, and correlate that with SA. These measures are generally unobtrusive and objective and can be used in conjunction with explicit measures. It has been suggested that these measures can succeed where explicit methods cannot, particularly in situations where a determination of the timing of events is important and where the subject may be unaware of his or her deficiency (Pritchett, Hansman, and Johnson 1995). These measures may examine performance at the task overall, or alter the task to determine whether the subject notices the change.

Subjective ratings are assessments of SA made either by an observer or by the subject. These ratings can use a number of different scales, and can be either direct or relative (such as by comparing SA in one situation with SA in another situation). Although these ratings can be affected by a number of factors, including task performance, one technique that has been studied extensively is the situation awareness rating technique (SART) (Taylor 1989). This technique has the operator rate 10 constructs on a seven-point scale. The constructs are grouped into three categories: (1) attentional demand (which includes the instability, variability, and complexity of the situation), (2) attentional supply (including arousal, spare mental capacity, concentration, and division of attention), and (3) understanding (which includes information quality and quantity, and familiarity). SART has been found to be more sensitive than overall subjective measures of SA (Selcon and Taylor 1989), although some researchers feel that SART confounds SA with workload (Jones and Endsley 2000).

33.2 FROM THE EARLY YEARS TO TODAY: APPLYING HUMAN–COMPUTER INTERACTION PRINCIPLES TO AVIATION

In addition to research concerning information display and assessment, much was being learned about automation in general. The Fitts report (1951b) mentioned the need to understand how tasks should be allocated between human and machine, and this was the subject of some effort on the part of Fitts himself. Computer technology was in its infancy, and little was known in the 1950s about what limits there may be on the ability of machines to assist humans. Many thought that these machines would gradually replace all functions of the humans. Researchers now understand better the different ways in which we can describe the relationship between human and operator, although this relationship is changing as new types of automated assistance are being added to aircraft and air traffic systems.

This section will discuss the concept of allocating roles to automation and humans, beginning with some of the early concepts, then progressing to more recent characterizations of the roles of humans and automation. Following this will be a discussion of how cognitive engineering has helped shape early systems, consisting mainly of control automation, and how it is significantly involved in newer types of systems, consisting mainly of information automation.

33.2.1 Function Allocation Principles

The "Fitts list" shown in Table 33.1 served as the launching point for studies on the allocation of tasks to humans and automation. Automation for the control of aircraft had been around for decades by 1951. However, not much more was automated until the late 1960s and early 1970s, when some systems, navigation, and automated alerting were added. The early 1980s saw the advent of graphical displays, flight management systems, and the removal of flight engineers. The late 1980s introduced the concept of glass cockpits—replacing nearly all electromechanical "steam gauges" with graphical displays (Billings 1997).

TABLE 33.1
Fitts List

Humans Are Good At	Machines Are Good At
Sensory functions and detection	Speed and power
Perceptual ability	Routine and repetitive work
Flexibility and improvisation	Computation
Judgment and selective recall	Short-term storage
Reasoning	Simultaneous operations
Long-term memory	Short-term memory

Over the decades, there have been several different attempts at methodologies for allocating function. This section will be organized around those efforts: the Fitts list, the "automate everything" philosophy, supervisory control, guidelines for automation, and levels of automation.

33.2.1.1 Fitts List

To discuss the advantages and disadvantages of this approach, but without rehashing Fitts' discussion of them, we will first discuss the contents of the list, and findings surrounding the truth or fallacy of the claims. Then, we will give examples of applying the Fitts list to automation, examining the strengths and weaknesses of the approach. Finally, examples of criticism levied at this method of function allocation will be discussed.

Many of the items in the list still appear to be true. Very little argument can be made with the assertion that machines can generate more power than humans, and they can perform many tasks faster than humans. On the other side of the list, humans are still much more capable of improvisation and making judgments, and computers are not able to reason in any meaningful way, despite significant advances in computer technology.

Yet, some comparisons appear to be more complicated than suggested by the Fitts list. Although a human's perceptual ability is quite impressive, there are machines capable of remarkable feats of detection—thermal imaging systems that can "see" through walls, satellites that can read individual license plates, microscopes that can image individual atoms, explosive detection equipment that can "smell" tiny amounts of specific chemicals, and so on. Despite this, it still requires a human to interpret the results. Machines still have only primitive (and brittle) abilities to deal with recognition (particularly of symbol systems, faces, and expressions, etc.).

In general, it appears humans are "wired" for rapid operation in a highly uncertain and diverse environment, whereas machines are still generally relegated to rather specific operations in a well-regulated and defined environment. So while machines may be better at routine and repetitive work, they are generally unable to deal with events or circumstances outside of the expected operating regime. Humans, however, are able to operate in the face of such events, leading to the notion that humans are suited for "supervising" automation, which will be discussed in Section 33.2.1.3.

In practice, Fitts lists have been difficult to apply in allocating function (Sheridan 1998). One reason is likely that a Fitts list approach requires automation to be assigned to a particular function. In reality, most functions require some aspects to be done by human operators and some part to be done by the automation. Since the functions are shared between humans and automation, any distinction of human or machine for a function is likely to be sub optimal.

33.2.1.2 The "Automate Everything" Philosophy

Although this method of allocation seems unjustifiable on its face, it has had its proponents, and its reasons. Humans typically have been seen as the source of a majority of error in the system. Replacing them with "trustworthy" automation, which also came with the benefits of lower cost and more efficient operation, did not seem quite so ridiculous several decades ago. Officials of two different aircraft manufacturers have been quoted as adhering to some form of this approach relatively recently (Billings 1997). In addition, these thoughts came at a time when the growth of technology appeared to be facilitating the design of automation.

It was becoming apparent by the late 1970s that humans and automation had trouble getting along. Among the problems frequently cited were (Wiener and Curry 1980) as follows:

- Crew errors exacerbating automation errors
- Improper setup of automated systems
- Improper response to alerts
- Failure to monitor
- Loss of proficiency

In truth, however, the hurdles for an entirely automated aircraft have been, so far, too great to be surmounted. For one, the current social and political climate would have to undergo significant changes. Furthermore, it has become apparent that automation is not as reliable as had been believed at the time. Examples of failures in automatic equipment have been common and persistent (Wiener and Curry 1980).

In addition, under this strategy, the operator would be left with any functions that could not be automated (Parasuraman, Sheridan, and Wickens 2000). The result is that ill-defined, difficult tasks may be assigned to an operator that is significantly detached from the operation of the system. It is unlikely that the operator could perform well under these circumstances.

In such a system, the operator may be left with the task of monitoring the system, with the resumption of control should the automation fail. Unfortunately, the skills necessary to do so would likely be degraded from nonuse. In addition, retrieval from long-term memory is more difficult if information goes unused for long periods of time, and the model of the system required for diagnosis and action is unlikely to be present in an operator who rarely actively controls it. These last few points have been referred to as "ironies of automation," since the more advanced the automation, the more valuable may be the operator's function, but the less able the operator may be to fulfill that function (Bainbridge 1983).

33.2.1.3 Human Supervisory Control

About the same time as automate-everything approach, a concept of interaction between humans and automation was being discussed, called human supervisory control (Ferrell and Sheridan 1967). This was an offshoot of research into how humans could control a remote vehicle (specifically vehicles on the moon). It became apparent that under the 3-second delay for information to be sent to and return from the moon, that some control loops would have to be closed by the automation, with the operator acting as a supervisor of the automation (Sheridan 1992).

The concept reflected the growing role of automation, and the changing of the human's role from operator to manager of automation. As shown in Figure 33.2, several levels of supervisory and control loops exist, with the human closing the supervisory loops, and each control loop being able to be closed by either the automation or the human.

Designers could now use control theory concepts to the human-automation problem. The role of humans was essentially defined by this paradigm, and the designer could then look at lower level tasks and turn to questions such as what human resources are required for each task, how much time

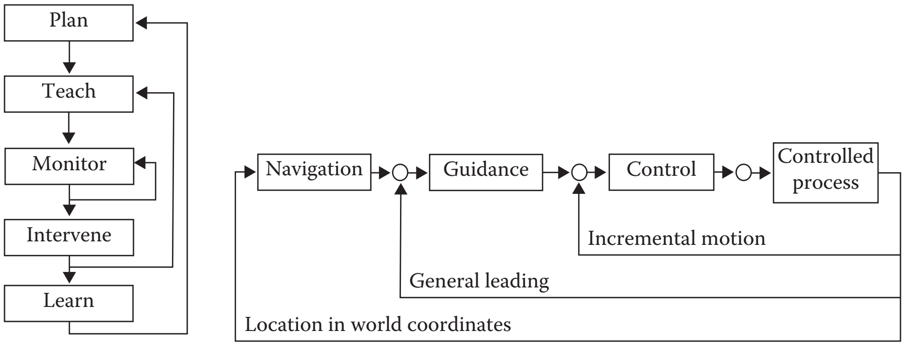


FIGURE 33.2 Sheridan's supervisory and control loops. (Adapted from Sheridan, T. B. *Telerobotics, Automation, and Human Supervisory Control,* Figures 1.2 and 1.3 © Massachusetts Institute of Technology, published by the MIT Press, 1992.)

and effort is required, how much time and energy is available, and what the consequences are of accomplishing or not accomplishing an action (Sheridan 1992).

Sheridan goes on to describe the limitations of humans and automation, and the implications of this for applying the concept. However, these admonishments went unheeded and automation was steadily added to the flight deck and control rooms, with the expectation that the operators would handle the additional managerial workload.

33.2.1.4 Levels of Automation

Parasuraman, Sheridan, and Wickens (2000) proposed a new model for allocation of automation functions across four classes and along a continuum of low automation to high automation. The four classes are: (1) information acquisition, (2) information analysis, (3) decision and action selection, and (4) action implementation. The level of automation can be allocated for each of these four levels, and can be adapted "on the fly" during operational use. The overriding consideration in determining level of automation is the consequences for human performance in the wake of that automation decision.

Information acquisition automation relates to the "sensing and registration of input data." This includes such things as moving sensors (low level of automation), organizing information (moderate), highlighting information (moderate), and filtering information (could be a high level of automation).

Information analysis automation relate to the requirement to integrate and/or transform information. Low-level examples include showing track predictions or trends. At higher-levels data may be integrated, displaying only high-level information to the operator.

Decision and action selection automation assists in determining courses of action and in making decisions. Low-level automation in this class may simply recommend courses of action. At higher levels, the automation may perform the action that it feels is appropriate. The difference is demonstrated by the Ground Proximity Warning System (GPWS), which recommends an action and is at a fairly low level, and the ground collision avoidance system, which is at a significantly higher level as it automatically takes control to avoid a ground impact if the pilot does not take action.

Action implementation automation performs actions for the operator. The level is determined by the amount of manual control left to the human operator. For example, an autopilot on heading hold may still require heading changes using a turn controller and speed control using throttles. This would be a lower level of action implementation than a fully coupled autopilot.

The appropriate level of automation for different classes should then be decided upon by examining possible solutions at different levels of automation and applying primary and secondary criteria in a possibly iterative fashion. The primary criteria are based on aspects of human performance such as mental workload, SA, complacency, and skill degradation. The secondary criteria are based on more practical concerns such as reliability of the automation and the costs versus benefits of automation decisions.

33.2.1.5 Adaptive and Adaptable Function Allocation

In addition to the other automation-related deficiencies mentioned in Section 33.2.1.2, it was noted that when certain functions of pilots were automated, those pilots suffered from increased complacency and reduced SA. To combat these problems, researchers investigated a concept whereby functions could be allocated dynamically between pilots and automation. This concept is referred to as "adaptive" function allocation when the allocation is controlled by automation, and as "adaptable" function allocation when the human operator controls the allocation (Miller and Parasuraman 2007).

Automation can alter the allocation of functions by monitoring physiological signals that reflect the workload, stress level, or engagement of the operator; by monitoring the performance level of the operator; or by reference to a model of the task. Based on a change to one of these levels, functions may be wholly allocated to automation (complete reallocation), allocated to a predefined level of automation (partial reallocation), or moved up or down by one or a few levels of automation (partial transformation) (Lagu and Landry 2011). Each of these methods has been used in research settings, although no operational dynamic function allocations have been implemented.

A pilot's general workload or stress level can be inferred from changes in heart rate or the interbeat interval (Rani et al. 2002; Wilson and Russell 2007). A pilot's cognitive workload can be inferred from changes in heart rate variability (Acharya et al. 2006). A pilot's engagement can be inferred from electro-encephalograph measurements (Bailey et al. 2006). Triggers for allocating automation to and from automation can be established on the basis of these measurements.

Performance-based measures either monitor overall performance or performance at particular, critical tasks. Should operator performance drop below a threshold, this is used as a trigger to allocate functions to the automation; if performance increases above the threshold, functions are allocated back to the human (Parasuraman et al. 1993). Critical task methods have been found to be insensitive to overall performance (Parasuraman and Byrne 2003).

Models of a task may be used to predict when workload is exceeding the capability of the operator, and when the operator may be under-loaded. These predictions can be used to trigger allocation between human and automation. A study of model-based triggers failed to find any difference between model-based and performance-based triggering methods (Parasuraman et al. 1993).

33.2.1.6 Ecological Allocation

Advocates of this approach (although they do not use this term to describe it), regard function-based representations of work as inadequate representations of the actual workplace. The workplace is complex, and there is interplay between technology and work—that is, just as the workplace is changed by automation, the use of the automation may be changed by the workplace. The claim is then that one cannot adequately allocate function without fully comprehending the work environment and the situations that arise therein.

Suchman (1987, p. 28) stressed that work is "situated," and as a result the system must "incorporate both a sensitivity to local circumstances and resources for the remedy of troubles in understanding that inevitably arise."

Moreover, the situations that arise can fundamentally change the work environment. In a discussion of previous work, Wright, Dearden, and Fields (1998) related three themes that emerged from their (and other related) studies:

> The first is that divisions of labor are often dynamically reallocated on the basis of local contingencies on an occasion of practice. The second is that the processes by which such working divisions of labor are achieved are a significant component of that work. The third themes (sic) is that even where divisions of labor are precomputed and reified in such mechanisms as standard operating procedures, there is still work to be done in making those procedures work (p. 339).

Often, due to the changing nature of work, organization, and even situation, the underlying functions of automation can change. In a study of two incidents in the London Underground (subway system) in which trains left their drivers behind, it was discovered that the operators had altered one of the two controls required to operate the train (Wright, Dearden, and Fields 1998). One automated control requires the doors of the train to all be shut before the train will move. A second control is simply a button for the operator to push to start the train moving. The operators, apparently believing that the second control was redundant, taped the button in the on position. The two incidents happened when operators left their station to fix stuck doors. When the doors were closed, the "circuit" was closed, and the trains began moving.

Wright, Dearden, and Fields (1998) argue that the operators had unknowingly changed the allocation of functions. Their function had been removed—the automation now had sole control to operate the vehicle. Yet, these types of changes to function are not only the result of errors. Pilots make use of automation in ways that change their function as well—stall warning systems as angle of attack indicators, situational awareness instruments as navigation sensors.

For this reason, it may be desirable to allocate functions dynamically, as it is done in normal social work contexts. In this concept, the (adaptable) allocation of function would depend on the context of each occasion of use (Scallen, Hancock, and Dudley 1996). In this case, designers would have to examine the consequences of improper configuration of the automation and how to mitigate the likelihood of that occurring (Dearden, Harrison, and Wright 1998).

One method of obtaining the richer representations of the work context called for in this concept would be the use of scenarios (Dearden, Harrison, and Wright 1998), as discussed in Chapter 54. Other methods include ethnographic approaches (Hughes, Randall, and Shapiro 1992) and contextual design methods (Beyer and Holtzblatt 1998), which are discussed in detail in Chapters 51 and 50, respectively. The former finds researchers immersing themselves in the work environment under study to attempt to understand the social organization of the work. In the latter, the work environment is viewed from different perspectives (e.g., user, cultural relationships, sequences of actions, information artifacts, physical layout) to define roles and tasks.

33.2.1.7 Guidelines for Function Allocation

Over the years, a number of efforts at establishing guidelines for automation have been described. Several of these, by notable authors are discussed here.

Wiener and Curry (1980) first presented guidelines for automation. They advised designers to create automation that was comprehensible to the operator, so that the operator could detect and properly react to failures in the system. They also advised that the system should appear to perform the task in much the same way as the operator, allowing for individual styles of operation while not affecting overall system performance. They suggested that operators should be well trained on the setup of the system as well as nominal and off-nominal operation, emphasizing its use as an aid rather than as a replacement. Operators should also be trained on the alerts of the systems, which should be able to be validated by the operator, and should clearly indicate both the source of the alarm and the severity of the situation. Wiener and Curry also suggested that monitoring may be an important task for the operator, who should be trained and motivated to monitor, and whose workload should be kept at a moderate level to avoid both high- and low-workload situations.

Billings (1997) agreed with many of these findings, but stressed that the automation should never have control over the system. The operator should be in control and even should not be automatically prevented from exceeding normal operating limitations. Billings also warned against overautomation, the tendency of complex automation to become incomprehensible, and the resulting system being too complex for the operator to control in the case of automation failure.

Parasuraman and Riley (1997) added to these guidelines by pointing out the possibility that in certain situations operators can "over-rely" on automation, whereas in other situations operators can "under-rely" on the automation. These situations generally occur when the operator does not have sufficient knowledge about the automation or the situation, or when confidence in the automation is misplaced (either high or low). They found that sudden alerts do not adequately allow the operators to prepare to respond, and suggested the use of preparatory, or "likelihood" alerts. They warned that automation in many cases replaced operator error with designer error, which can occur if designers do not incorporate knowledge of operational practice into their design, and do not allow for some nonintended uses of the automation.

33.2.2 Autopilots

The autopilot's development began in 1914 when Lawrence Sperry created a gyroscopic device capable of stabilizing an aircraft's flight without input from the pilot. This lead to the design of a fully automatic aircraft by 1915, and a fully automatic pilot by 1916, although it would not be until 1931 that an autopilot was licensed to operate with passengers on board (Davenport 1978). An automatic landing system was invented in 1937 by Capt. Carl Crane, and since then there have been numerous refinements in control theory which have been applied to autopilots.

One particularly interesting addition to control theory was the finding that the behavior of human–machine systems with simple feedback control systems (such as an aircraft under nominal flight conditions) can be modeled using a simple control law (McRuer and Graham 1965). McRuer attempted to model just the pilot as a controller of the system, in a manner consistent with the development of automatic controllers, such as the autopilot. Such automatic controllers have a mathematical transfer function that outputs control movements when provided with deviations of system state from the desired state as input. However, McRuer could not construct such a mathematical model of the pilot. Instead, a simple delay-and-integrator control system model was found for the combination of the system and operator; because the pilot could adapt to the dynamics of the system, the control function of just the pilot was inseparable from the plant dynamics. As stated by Sheridan, "the right way to model the human was to model the whole closed loop: the human and the airplane, or whatever system he was controlling, together as a single entity ... this was a great insight" (Gerovitch 2003).

In general, autopilots have been amazingly successful. They relieve pilots of a great deal of manual workload, particularly during routine phases of flight such as cruise. Autopilots are also capable of increased accuracy at navigation, which is reflected in their ability to land aircraft in poor weather, including when poor visibility completely obstructs the pilot from seeing the runway. Autopilots are an excellent example of the successful introduction of automation to replace manual work in a complex human-integrated system.

However, autopilots are not exempt from HCI issues. Most notably, pilots may experience confusion regarding the mode of the autopilot; the concept of "mode confusion" is discussed in Section 33.3.1.

However, in one sense the challenge of introducing autopilots is not as great as the next generation of challenges. The autopilot automates a task that the pilot can still perform; if the system fails, the operator can detect the failure and resume manual control. In this way, it is much like the cruise control in an automobile. In future systems, it is desired to introduce automation to allow the system to perform beyond the limits of manual control. In such systems, operators will not be able to detect failures or intervene should the system fail. (Such control systems exist for aircraft that are inherently unstable, such as the X-29 and the Eurofighter.)

33.2.3 Aviation Checklists and Procedures

Aviation operations are highly proceduralized, and nearly all aspects of the operation of an aircraft involve the use of checklists. In safety critical environments such as aviation, such proceduralization is common. Recognizing this, engineers who design flight deck instrumentation consider procedures when developing new avionics. However, there has been relatively little human factors work on the design and utilization of the procedures themselves (Degani and Weiner 1993).

Among the work that has been done, general design principles and considerations have been described by Degani and Weiner (1990, 1997). De Brito (2002) and Endsley and Garland (2000) found that experience will influence whether and how pilots follow procedures. However, the specific effect is not straightforward, as inexperienced pilots tend to deviate from procedure improperly, whereas experienced pilots tend to deviate from procedure when the circumstances dictate that the procedure is not appropriate. Ockerman and Pritchett (2004) recommend that the detail of the procedure is limited to force experienced operators to rely on their knowledge of the situation, and that the boundary conditions of the procedure are explicitly presented. Findings from both aviation and other domains indicate that enforcing compliance to procedures through management action may lead to worse performance (Dien 1998; Graeber and Moodi 1998; Parker and Lawton 2000).

33.2.4 Air Traffic Control

Efforts at controlling air traffic began quickly after the advent of aircraft. These early efforts consisted of drafting procedures for safe operation of aircraft, and were undertaken by private companies until the 1930s (Gilbert 1973). As aviation grew, the efforts undertaken by dedicated professionals to track aircraft and ensure their separation from one another increased.

This process was, and to a surprising degree still is, a highly manual process. Initially, aircraft had contact via radio with their companies, who relayed position and timing information to a central control facility. The controllers in that facility would record this information on a large blackboard, and manipulate aircraft markers (also called "shrimp boats") on a map table. They would analyze this information in an attempt to predict potential conflicts between aircraft. This ability to mentally project aircraft states is still relied on today when radar displays fail.

Progress from these modest beginnings was slow compared with the progress of aircraft technology. The two most significant advances in air traffic technology occurred after World War II. At least partially due to their experiences in the Berlin Airlift, air traffic personnel adopted the practice of speaking directly to aircraft. In addition, the development of radar allowed controllers to track aircraft in real time.

33.2.4.1 Early Radar Displays

Although the application to air traffic control is obvious, early efforts at controlling aircraft using radar were troubled. Early radar displays displayed the position of each aircraft relative to the radar station, but did not display other critical information, such as aircraft identification, speed, or altitude. Because of this, associating targets with radar returns was difficult, and projecting their status into the future was still a largely cognitive process. The recognition of this helped bring about the introduction of secondary surveillance radar, in which a transponder on board each aircraft transmits an identification code and altitude to a ground receiving station (Gilbert 1973). This information can then be correlated with the radar returns, eliminating the need for map tables.

33.2.4.2 Flight Strips and Strip Bays

Controllers gradually replaced the blackboard with small strips of paper on which information about each flight was recorded. These would be placed in a rack (or bay) in temporal order, and even passed from controller to controller as the flight progressed. As computers were added in the 1960s and 1970s, one of their first tasks was to automate this flight data processing, including printing flight strips.

This proved to be a rather difficult task. The physical manipulation of the flight strips, such as controllers moving the strip from one position to another or literally handing it off to another controller, contained significant meaning (Bentley et al. 1992; MacKay 1999). This meaning was lost when the strips were no longer physically present. Since this time, data strips have been automated within the User Request Evaluation Tool (URET), but the manner in which the deficiencies were overcome, or the resulting performance of controllers, has not been documented.

33.3 MODERN CHALLENGES, SOLUTIONS, AND PRINCIPLES

The first 50 years of aviation saw significant changes and innovations in both aircraft and air traffic control. Recent developments, such as complex flight management systems, multifunction displays, highly integrated information systems, alerting and warning systems, and sophisticated automatic control capability have presented even more complex challenges for aviation professionals and researchers.

33.3.1 Glass Cockpits, Flight Management Systems, Mental Models, and Mode Confusion

Beginning in the 1970s, airliners were being equipped with flight management computers, which would assist pilots in numerous tasks, including flight path management, navigation, and fuel burn management. The interface to the flight management computer is through a control display unit, which consists of a small alphanumeric keyboard, several function keys, and a small display screen, all of which fits into about a 6 inch by 10 inch space. On this unit, pilots must enter navigation information, select information to be displayed, and navigate through menus. By the 1980s, flight management computers were integrated with CRT displays, creating flight decks mostly devoid of gauges, also called “glass cockpits.” An example of a glass cockpit (from NASA’s Atlantis orbiter) is shown in Figure 33.3.

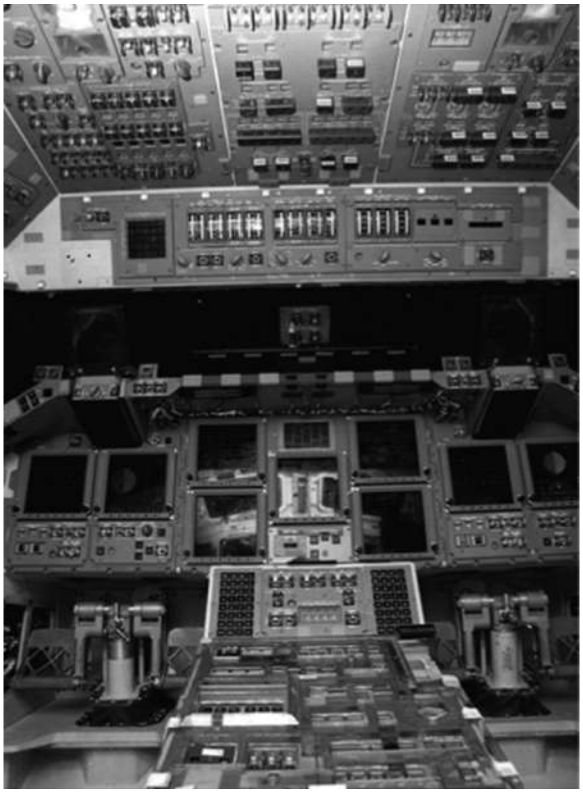


FIGURE 33.3 Orbiter glass cockpit. (Image courtesy of NASA, obtained from http://www.nasa.gov/mission_pages/shuttle/main/shuttle_evolves.html.)

Such systems integrated the operation of the aircraft with navigation, resulting in a “fundamental shift in aircraft automation” (Billings 1997, p. 107). Although such displays provided the benefits of a highly integrated display, there is a cost in terms of workload, crew coordination, error management, and vigilance (Wickens et al. 1998). In addition, a number of accidents have been associated with pilots misunderstanding the state of the automation. These accidents have been attributed to the pilots lacking an accurate “mental model” of the automation.

Mental models, described in detail in Chapter 4, are a conceptual representation of a system posited cognitively. The precursors to mental model research date back nearly a century. These early inquiries (Craik 1943; Wittgenstein 1922), and a significant body of recent work, argue that cognition is inherently imagistic, so that mental models are a theory of how people think. This theory is a topic of contentious debate in the psychological community, with many researchers vehemently denying that people think in these “pictures in the head,” but rather that thought consists of formal rules of inference (Pylyshyn 2003).

Undeniably, however, people at least think about such imagistic models. In doing so, operator’s reason about their system based on these models. There is convincing evidence that such models correlate well with certain operator errors, and point to ways in which good design can mitigate these errors. As such, we will try to skirt the contentious issue of whether mental models are theories or models, and stick to how the concept has been deployed in support of cognitive engineering.

Mental model research was borne out of the observation that many people use analogy to understand complex phenomena. This idea became more formalized in the cognitive science literature, highlighted by a flood of papers and books on mental models in the early 1980s (Gentner and Stevens 1983; Johnson-Laird 1983). If one believed that a person's knowledge of a complex system was a product of that person's model of the system, then an understanding of the limitations of these models was crucial to explaining behavior and errors.

Nowhere did mental models seem to fit in better than in explaining a number of human-automation interaction failures. Flight deck automation was becoming increasingly complex, taxing the abilities of pilots to accurately model it. Where such inaccuracies exist, error is likely. Several particularly prominent examples of this have occurred in the last 15 years. In 1988, a Boeing 767 was inadvertently put into a vertical speed mode, nearly resulting in a crash. A similar situation occurred in an Airbus 320, when a crew mistakenly selected a descent rate of 3300 feet per minute instead of on a 3.3° downward slope and flew into terrain. In 1994, a China Airlines Airbus 300 aircraft crashed after pilots tried to continue a landing while the autopilot had been inadvertently put into a go-around mode. Also in 1988, an Airbus 320 was flown into terrain by an aircrew during an air show; the crew believed that the envelope protection of the aircraft would prevent a stall, but the aircraft was in a mode where that protection was not provided.

These incidents indicated that one piece of flight deck automation particularly prone to such mental model inaccuracies is the flight management system, which interacts closely with the autopilot. Researchers had been aware of mode problems in HCI for some time, and in due time, these issues arose in the interaction of the pilot with the computer that operates the autopilot. This aspect of the problem—as a computer interface issue—has significantly affected the research on mental models in aviation.

In addition to calling for training for pilots to better understand their systems, much effort has been put into creating models of the pilot's interaction with the flight management system in an effort to understand how systems might be designed to mitigate these model inaccuracies.

33.3.2 Heads-up Displays

To minimize the need to view instruments inside the cockpit, "heads-up" displays were superimposed on the windscreen, providing at least the basic T information. The same method can be used to superimpose images on the inside of the visor of a helmet (known as helmet-mounted displays). HUDs have moved from the military, where they found their first application, to commercial aircraft and even automobiles.

One major difference between the instrumentation on a HUD and the same instrumentation on a traditional instrument panel is that the world can be viewed through the HUD. This allows for the possibility of integrating real external images with the instrumentation without violating the proximity principle. However, the flip side of this is that the world peeking through the HUD provides a great deal of clutter to the display, although some research indicates that this is not a significant increase in difficulty (Ververs and Wickens 1998). Overall, the HUD has a number of advantages that make it superior to conventional positioning of displays. The HUD enhances the operator's ability to switch between near and far domains (Levy, Foyle, and McCann 1998), although strangely this reverses to a detriment for the detection of unexpected events (Wickens and Prevett 1995).

33.3.3 Information Automation

The rapid rise in computer and display technology has allowed engineers to develop systems that alleviate some of the information burden on the operator. These systems perform automated monitoring (such as conflict alerting and prevention), provide information to assist the pilot's decision making, and help provide coordination between different operators.

33.3.3.1 Alerting Systems

In the late 1970s, aircraft were undergoing a rapid expansion in terms of warnings and alerting systems. One survey found general agreement that there were too many warnings and also standards and guidelines were needed for warning system design (Cooper 1977). Another researcher determined that the number of visual alerts and flags had doubled, whereas the number of aural alerts (see Chapter 10) had increased by 50% since the beginning of the commercial jet age (Veitengruber 1977). Based on the data at the time, researchers were recommending that alerts be reduced, prioritized, and inhibited during critical phases of flight.

Since then alerting systems have grown significantly in capability and complexity. Current alerting systems may utilize multiple sensors, ground communications, large knowledge databases, and may have sophisticated algorithms for alert threshold and guidance recommendations (Pritchett 2001).

One of the most widely analyzed problems in alerting system technology relates to problems with conformance to the GPWS. The GPWS was a complex alert designed to warn of something extremely hazardous—proximity to or closure to terrain. It was complex in that closure to terrain can be caused by a number of things, including landing, and the system had to discriminate these events. In part due to this complexity, and in part due to poor design, the system frequently produced false alarms. A high rate of false alarms generally has the effect of reducing operator trust in the automation and causing under reliance (the pilots would not believe the alert), a phenomenon known as the "cry-wolf" effect (Breznitz 1984).

High false alarm rates can also be caused by improper setting of thresholds for alerts, a process that is analogous to signal detection. The threshold for an alert can be based on the system operating characteristics curve, which is in turn a function of the sensors (Kuchar 1996). Setting the threshold is then a tradeoff between missed detections and false alarms. Typically, this means that the fewer the false alarms, the more the missed detections. Interestingly, some researchers have warned against having too low a false alarm rate (Farber and Paley 1993). If the false alarm rate is too low, not only will the occurrence be infrequent and unexpected, but it will likely require a later alert, compounding the difficulty of reacting to the alert.

The problem of false alarms is further exacerbated by the low base rate of incidents requiring an alert (Parasuraman and Riley 1997). In these cases, the relative rate of false alarms is increased. Yet in many cases, the cost of a false alarm is considered much lower than a missed detection, leading designers to err on the side of economy, and/or safety.

These (generally) are cases where the alerting system is functioning as a signal detector or a hazard detector, the latter being a more complex version of the former. Alerting systems have also added hazard-resolving logic, such as in Traffic Collision and Avoidance System (TCAS) II (Kuchar and Drumm 2007) and NASA's Advanced Airspace Concept (Erzberger and Paielli 2002). The benefit of such an approach is that the resolving maneuver can be incorporated into the alerting logic, resulting in a reduction of false alarms. However, such a system relies heavily on the pilot performing the maneuver appropriately. It is also possible that the maneuver may actually induce a collision, by creating a problem where none actually existed.

33.3.3.2 Decision-Support Systems

In addition to alerting systems, there are also systems on board aircraft and, to an even greater extent, in air traffic control which are considered decision-support tools (see Chapter 26 for a discussion of design approaches for decision-support systems). These tools do not warn or alert, but provide information to support decision making. This information is often integrated from several sources, or enhanced in some way.

One fairly simple example of a decision-support system on a flight deck is the weather radar system. This system provides no discrete alert, but rather shows the pilot where hazardous weather exists to help pilots determine how to deviate to avoid it. With no weather radar, pilots had to rely on visual means, which can only detect gross features, such as height, darkness, and the presence of lightning. However, weather radar not only shows rates of precipitation (an indication of severity), but can also show the lateral and vertical extent of the weather.

Another decision-support system is the Engine Indication and Crew Alerting System. Although some aspects of this system are more correctly considered an alerting system, much of the system is simply decision support. This display shows pilots the status of most of the systems on board. It can be configured to depict engine instruments or other graphical depictions of the operation of the systems on board the aircraft.

As mentioned, there are a number of decision-support tools being used in air traffic control. In fact, the number of such displays are increasing at such a rate that the FAA is attempting to integrate these systems to reduce the sheer number of separate systems operating in air traffic facilities.

33.3.4 Near Term Air Traffic Control Issues in Human–Computer Interaction

The current air traffic control system, while reliant on computer resources, has been designed to be resilient to the loss of such resources. Computers are still used for display of primary and secondary radar information, and flight data processing. The upgrading of these capabilities from the 1970s has been slow, impacted by the widespread replacement of experienced controllers following the air traffic controller strike of 1981, and the failure of the Advanced Automation System (AAS) program of the late 1980s (Lee and Davis 1996). On the heels of the AAS system, however, have come a number of advances, including attempts to automate flight strips and the introduction of automation aids.

33.3.4.1 Automated Flight Strips

Flight strips have proven remarkably resilient to change. Many controllers still use flight strips, and flight strip bays are still used in air traffic control facilities. Recently, however, manual flight strips are being replaced with electronic flight strips, most notably in the URET, which took 25 years to implement (Arthur and McLaughlin 1998). Similar efforts to replace flight strips were undertaken in Europe (Berndtsson and Normark 2000). The reasons for their slow demise are highly instructive of some of the problems with automating some forms of human work.

Despite the significant workload of the manual process of dealing with flight strips, controllers have been reluctant to part with them, or to use suggested replacements. This is, in large part, due to the flight strips contribution to the controller's SA. For a controller, whose job is still to a large degree cognitive, their SA is crucial (Whitfield and Jackson 1983). Researchers have found that controllers indicate that flight strips are a primary means of establishing and maintaining their SA (Harper and Hughes 1991).

33.3.4.2 Decision-Support Tools in Air Traffic Control

One of the most significant areas of improvement within air traffic over the last two decades has been in decision-support tools. Over the years, a number of automation tools have been slowly added to the air traffic control system, and others have been considered and failed. These advancements have included the introduction of two conflict prediction systems, the failed attempt to provide sequencing and runway selection aids, and the introduction of centrally scheduled time-based metering tools.

The first conflict prediction system was created in the early 1980s as a quality assurance program. The "operational error detection program" (OEDP) was designed to identify when two aircraft had come closer to one another than allowed under FAA regulations. If the OEDP was set off, a record was made of the transgression, and investigation took place. This is obviously highly undesirable for controllers, who have nicknamed the system "the snitch patch." Controllers routinely pad separation between aircraft, in part to ensure that the snitch patch does not activate (Cotton 2003).

The conflict alert system gives a 3-minute warning that a violation is going to occur, although the system is not considered highly reliable by controllers due to its high false alarm rate (Wickens et al. 2009). Part of the reason for the system's unreliability is that its predictions are based on simple extrapolations of the current states of the aircraft involved, without consideration of intentions (such as its intention to stop climbing or to change course).The system's unreliability has led to several efforts to replace it. The URET system mentioned previously includes a "conflict probe," which examines the predicted flight path of aircraft and gives an indication of the potential for collision (Brudnicki and McFarland 1997).

In the 1990s, NASA undertook an effort to provide automation tools for controllers and traffic managers. The suite of tools, called the Center-TRACON Automation System (CTAS), was based on combining radar position and speed data, aircraft flight plans, weather data, and models of aircraft flight characteristics. The models were based upon those used in the flight management systems in aircraft, which are known to be highly accurate. One of the first tools proposed was the Final Approach Spacing Tool (FAST). This system was to be used by controllers handling aircraft close to busy airports. These controllers transition aircraft from intermediate altitudes and about 50 miles out from an airport and coordinate their arrival to the runway. This is a very complex task, requiring controllers to predict the proper sequencing and merging of aircraft from several directions. FAST generated advisories for sequencing, runway assignment, headings, and speeds to ease the merging and spacing of aircraft (Lee and Davis 1996). Although deemed acceptable by controllers (Lee and Sanford 1998), the system was ultimately not implemented. One of the likely factors was that although the system was extremely accurate in its predictions, it was not perfectly so, and controllers may have felt that the workload and distraction of the few inaccurate predictions outweighed its benefits.

Another part of the CTAS suite of tools is the Traffic Management Advisor (TMA), which provides a similar service as FAST for controllers handling aircraft further out from the airport (out to about 300 miles). This system provides delay advisories for each aircraft approaching a busy airport. Controllers are then responsible for slowing aircraft to meet the assigned delays, utilizing the same techniques they use to space aircraft normally. The main shift between the current method and TMA is that the spacing TMA provides is time-based, versus the distance-based method currently used. Time-based metering in this fashion has been shown to be more efficient than distance-based (Sokkappa 1989; Vandevenne and Andrews 1993), and TMA has shown significant benefits where it has been adopted (Knorr 2003). Similar efforts focused on runway sequencing and scheduling have been developed in Europe, Canada, and Australia (Barco Orthogon 2003; Ljundberg and Lucas 1992; NavCanada 2003; Robinson III, Davis, and Isaacson 1997). More advanced systems are also being developed which move some separation authority to the flight deck (Prevot et al. 2003).

33.4 FUTURE CHALLENGES

Aviation has undergone a great deal of change since its inception, although predictions of its development path have not always been accurate. Forty or fifty years ago predictions of where the air transport system would be probably have included much more automation of air traffic control, high reliance on satellite systems, with larger and faster aircraft than were then utilized. These aspects have largely not been realized to the extent expected. The aviation system of the future will undoubtedly hold other unexpected developments, but a few challenges for researchers, engineers, and practitioners are clear.

33.4.1 Three-Dimensional Displays

To overcome the limitations of two-dimensional (2D) displays, several alternatives to providing three dimensions in a display have been investigated. Air traffic and navigation displays are both trying to represent three dimensions in a 2D display. Inevitably, some information is lost in the display in this case. This can be accomplished through binocular, stereoscopic, or holographic devices (Mountford and Somberg 1981), but practical problems make this unlikely. Also unlikely due to the cognitive demands of integration is using time-frame compressions (a display of sequential images) (Roscoe, Corl, and Jensen 1981). However, significant research has been accomplished in using perspective illusion displays, where the illusion of three dimensions is given through the use of perspective.

Three dimensional displays can portray a variety of viewpoints, to different effect (Wickens and Prevett 1995). In testing such displays, 2D displays, with either some analog display of altitude or an additional profile view, are used for comparison. This adds complexity to testing these displays, since often additional information has to be added to current displays to isolate the 3D presentation from the additional information provided by the display.

In addition, dual panel 2D displays, although they may contain the same information as 3D displays, have an additional cost—that of scanning and integrating. So, it would seem that 3D displays adhere to several tenets of good display design: proximity compatibility, integration, and pictorial realism.

However, if the airplane is not navigating vertically, the 3D display may impose a clutter cost as the operator must sort through the irrelevant altitude information. This is in addition to the problem of ambiguity of position of objects along the line of sight of the display (McGreevy and Ellis 1986). Lastly, if the 3D perspective is immersed (the perspective is as the world looks from the cockpit), no information concerning the world to the side or behind the aircraft is presented, resulting in a "keyhole" view of the world (Woods 1984).

For judgments requiring mapping between the world and the display, a 3D display with a viewpoint outside the cockpit seems to provide the best performance (Olmos, Wickens, and Chudy 1997). This benefit is again tied to the principles of realism and integration, and is generally viewed as providing better overall SA than immersed 3D views.

33.4.1.1 Tunnels and Tubes in the Sky

Enhancements to the immersed viewpoint of a 3D display can present the operator with a highly integrated pathway, or "tunnel in the sky." Superior performance in navigation and tracking has been demonstrated using this display (Wickens and Prevett 1995), although they are currently still in development. The biggest obstacle to the deployment of such automation seems to be the cost of certification rather than any technical or cognitive issues. It appears that these displays only suffer from effects similar to those found on other perspective displays (confounding of distance along the line of sight) and on HUDs (difficulty in detecting unusual events due to the compelling nature of the display) (Wickens 1998).

Proposals that would segregate airspace by the level of equipage of the aircraft in that airspace often use the notion of "tubes in the sky," where the tube is a region of airspace in which only aircraft that have particular equipment are allowed to fly (Sridhar et al. 2007). Tubes and other proposals, such as dynamic airspace reconfiguration (Kopardekar, Bilimoria, and Sridhar 2007), would have significant consequences for air traffic controllers. The effect of such proposals is not yet known.

33.4.2 Coordination Support Systems

One type of information system which is likely to become prevalent in the future is what is referred to here as a coordination support system (for a further discussion of computer-supported cooperative work, see Chapter 28). It is becoming more common that work, including in the air traffic system, is distributed across organizations. For example, alerts generated by an aircraft's TCAS system have to be coordinated between the two aircraft in the conflict pair to avoid having the aircraft's resolutions conflict. Also, coordinating arrivals into airports involves affecting aircraft many hundreds of miles away from the airport, across air traffic control organizational, and even facility boundaries.

One approach to enabling such cooperative work is to actively share information between organizations through procedural means. Such an approach means that one organization, with localized information of interest, distributes that information to other parties. However, in the air traffic system, the great majority of the information needed for collaborative work is present in the task environment, or relates to the impact of other organizations' decisions on the traffic situation. If such ecological information could be provided, then collaborative work would happen naturally.

Such an approach is supported by the concept of a common information space (CIS) (Schmidt and Bannon 1992). A CIS is a source of information shared within and across organization boundaries to conduct coordinated work for common purposes. Instead of passing information from one organization to another, information is made available through the CIS, which can then be accessed and manipulated by any interested party. Moreover, objects can be created within the CIS that serve as translators across different communities. These objects, referred to as boundary objects, are entities in which decontextualized information resides in a particular form (Starr and Griesemer 1989). The distinction which defines boundary objects is their purpose in acting as translators, or, as defined by Bannon, they are a "means for sharing items in a common information space" (Bannon 2000, p. 5). Information, devoid of context, is presented in a form that allows users from different communities of interest to comprehend it within their own context. This object is accessed by users in different communities of interest to conduct their particular portion of the work. The information in boundary objects is contextualized only within each community of interest, and this context differs in each separate community.

33.4.3 Technologies Needed for 3x Air Traffic: Human–Computer Interaction Issues

The rate of growth of air traffic has been consistent for a number of decades, and is not expected to change significantly in the next few decades. Such a rate of growth would double or triple air traffic, a level which some researchers feel is unsupportable by the current method of air traffic control.

Such a conclusion is consistent with the historical development of the air traffic system. To grow beyond its capabilities in the 1940s, radar was added, reducing the workload of the controllers, which enabled them to deal with more traffic and higher aircraft speeds. To continue to grow, computerized flight data processing was added, further reducing workload and allowing controllers to handle more traffic. Increases in staffing, reduction in sector sizes, strategic control, and other innovations have allowed further increases in air traffic levels. To further continue these increases, researchers argue that a more fundamental shift is needed, by sharing some of the air traffic responsibilities between the current authority (air traffic controllers) and the flight deck, by allowing automation to accomplish some current tasks of the controllers, or some combination of these.

33.4.3.1 Data Communications

To date, much of the communication that occurs in the air transportation system is by voice, with the exception of communications between airline operations centers and their aircraft, which happens using a form of electronic mail. Voice communication is considered the primary means of controlling aircraft; controllers worry more about losing voice communication than radar. Specific procedures have been developed for aircraft to follow should communications be lost.

However, voice communication channels have limited bandwidth. The commands a controller issues to a pilot are stated in clear, standardized phraseology, and each command must be "read back" to the controller to ensure it has been heard correctly. Frequencies can become jammed with such communications, which is why commercial aircraft are required to simultaneously monitor the air traffic control frequency and an emergency frequency. Controllers in very busy areas, such as the airspace near a major airport, can encounter "frequency congestion," where the controller no longer waits for the read back, but instead issues streams of commands to different pilots back-to-back without pause. In such conditions, the capacity of the airspace is effectively capped to the number of aircraft with which the controller can communicate.

Because of this, and due to problems associated with miscommunication, it has been proposed to replace some, or even most, voice communications with data communications (Erzberger 2004a; Kerns 1991). In some early studies, however, controller's responses were slower when using data communications (Latorella 1998; Wickens, Miller, and Tham 1996). In addition, pilots were found to use information from communications to other pilots, referred to as "party line" information, and this information would be lost if the switch were made to data communications (Midkiff and Hansman 1993). Nonetheless, due to the need to increase capacity, it is generally assumed that data communications will replace at least some of the voice communications currently in use (Djokic, Lorenz, and Fricke 2010).

33.4.3.2 Four-Dimensional Contracts: Future Changes in the Roles of Pilots and Controllers?

The flight management systems used by the current generation of commercial aircraft is capable of meeting arrival times over points in its flight path, subject to the constraints of the flight envelope. The logic behind these flight management systems is also available to ground-based systems, and is in fact the heart of CTAS, a suite of air traffic management tools developed by NASA (Erzberger and Nedell 1989).

These ground-based systems can use the flight management algorithms to make predictions of arrival times for all aircraft in the National Airspace System (NAS). This has been accomplished for flights arriving at a number of the busiest airports in the United States in the TMA system (Swenson et al. 1997). This capability is being expanded to regional or national airspace, and for scheduling aircraft over any congested point in the NAS (Landry, Farley, and Hoang 2005).

One proposal for increasing the capacity of the air traffic system is to allow such a system to negotiate "4D contracts," indicating navigational waypoints, altitudes, and times of arrival at those points for all aircraft. The flight management systems would then be responsible for adhering to the contract. These contracts would ensure conflict-free routes, and coordinate arrivals into congested resources, including airports.

Such a system would radically alter the roles of pilots and controllers in relation to their automation. Pilots and controllers would be monitors of a highly automated system, in comparison with their current roles of actively controlling the flight paths of aircraft. As such, this presents enormous challenges to the aviation HCI community.

33.4.3.3 Automating Separation Assurance

The primary responsibility of an air traffic controller is to provide separation assurance between aircraft. Even if data link and 4D contracts were established, the controller is still limited by the cognitively intensive process of monitoring for conflicts. This task is also under consideration for automated assistance, either on the flight deck or through a centralized system.

One proposal is to equip flight decks with a Cockpit Display of Traffic Information system, and utilize an advanced form of the current TCAS. A second proposal is to utilize a centralized system for providing alerts (Erzberger 2004b). Such a system would coordinate resolutions automatically, and provide for an independent backup in case of failures to resolve. In both cases, automation would detect conflicts, identify resolutions, and (perhaps) automatically perform the resolutions. Such systems overcome the latencies in the human detection, communication, and action cycles. However, as the human operators are pushed back further from the actual operation of the systems, their ability to intervene in the case of automation failures wanes. Because of this, such automation must be highly reliable for the system to succeed.

33.4.4 Uninhabited Aerial Vehicles in the Airspace System

The rapid development of uninhabited aerial vehicles (UAVs) will also greatly impact the air traffic system of the future. To date such vehicles have been mostly constrained to special uses, such as the military. However, it seems likely that these vehicles will need to be integrated into the air traffic control system as their uses expand (Wegerbauer 2005).

The difficulties in introducing UAVs into the airspace system, from the perspective of the air traffic controller, include their unpredictable response to system failures and the substantial difference in performance between UAVs and commercial aircraft. As mentioned, procedures exist for cases in which systems failures, such as radio failure, occur. If such a failure occurs, controllers can manage the other aircraft assuming that the procedure is followed. Even under unusual circumstances, controllers can usually predict the behavior of the pilot and control other aircraft accordingly. However, if radio links to the UAV are lost, control of the vehicle would resort to a set of automated routines, which may not perform in a predictable manner. Moreover, without the ability to "see and avoid" other aircraft, UAVs pose a collision risk in such circumstances.

From the pilot's perspective, flying a UAV is substantially different than flying an aircraft. Many subtle, but important, proximate cues available to the pilot of a manned vehicle, such as vestibular and auditory cues, are not available to the pilot of a UAV. Instead, all information available to the UAV pilot is mediated by a computer display and data communications link, which may include some delay.

In addition, automatic landings, and landings using video and instrumentation, are not available for all UAVs. When visually landing such aircraft, the perspective is no longer egocentric, as with a pilot on the flight deck, but rather exocentric and from different perspectives, such as in front of the aircraft, behind it, or off to the side. These different perspectives make precision control of the vehicle, generally necessary for landing, difficult. As a result, accident rates for UAVs are high; although this does not usually involve loss of life, there is substantial cost in terms of material.

33.5 CONCLUSIONS

Fitts (1951b) and his colleagues described the research challenges in visual displays as of 1951. The document is remarkable in that it spurred (or foresaw) a great deal of research that continues to this day. Some of that can be attributed to the general nature of the questions, but many of the specific research questions and methods have been examined and used over the last 40 years. What was noticeably (and understandably) absent from the discussion of transitory displays was any reference to higher cognition. In some ways, this is indicative of the progress made in aerospace human–machine system research. Many of the difficulties associated with these concepts originated as higher levels (and greater amounts) of automation were added to the flight deck.

The progression of research has also followed this model. Early research resulted in a number of principles for integrating displays, for determining the form of displays for different purposes, and for the positioning of displays. Later work has concentrated on higher level cognitive issues such as mental models and SA. Also, work has been done on understanding how to apply automation, resulting in a number of principles regarding automation use and allocation of function. Most recently, new types of displays and automation such as warning systems and decision-support systems have been introduced, yielding a new set of problems for researchers to tackle. As the air transportation system matures and the demand for ever greater levels of capacity increases, it seems that greater reliance on automated systems will be required. For such a system to be successful, such automation must be based on sound principles.

REFERENCES

Abbott, T. S., G. C. Mowen, L. H. Person Jr., G. L. Keyser Jr., K. R. Yenni, and J. F. Garren Jr. 1980. Flight investigation of cockpit-displayed traffic information utilizing coded symbology in an advanced operational environment (NASA Technical Paper 1684). Hampton, VA: NASA Langley Research Center.

Acharya, R., P. Joseph, N. Kannathal, C. M. Lim, and J. S. Suri. 2006. Heart rate variability: A review. *Med Biol Eng Comput* 44:1031–51.

Allen, R. B. 1983. Cognitive factors in the use of menus and trees: An experiment. *IEEE J Sel Areas Commun* 1(2):333–36.

Arthur, W. C., and M. P. McLaughlin. 1998. *User Request Evaluation Tool (URET): Interfacility Conflict Probe Performance Assessment*. McLean, VA: MITRE.

Bailey, N. R., M. W. Scerbo, F. G. Freeman, P. J. Mikulka, and L. A. Scott. 2006. Comparison of a brain-based adaptive system and a manual adaptable system for invoking automation. *Hum Factors* 48:693–709.

Bainbridge, L. 1983. Ironies of automation. *Automatica* 19(6):775–80.

Bannon, L. J. 2000. Understanding common information spaces in CSCW. Paper presented at the Workshop on Cooperative Organisation of Common Information Spaces, Delft.

Barco Orthogon. 2003. Osyris. Retreived January 27, 2005, from http://www.barco.com/barcoview/downloads/BVW_Osyris_new_6p.pdf.

Bartlett, F. C. 1943. Instrument controls and displays – Efficient human manipulation (No. 565). London: UK Medical Research Council.

Bejczy, A. K., and G. Paine. 1977. Displays for supervisory control of manipulators. Paper presented at the 13th Annual Conference on Manual Control. Cambridge, MA.

Bentley, R., J. A. Hughes, D. Randall, T. Rodden, P. Sawyer, D. Shapiro et al. 1992. Ethnographically-informed systems design for air traffic control. Paper presented at the Computer Supported Cooperative Work, Toronto, Canada.

Beringer, D. B., R. C. Willeges, and S. N. Roscoe. 1975. The transition of experienced pilots to a frequency-separated aircraft altitude display. *Hum Factors* 17:401–14.

Berndtsson, J., and M. Normark. 2000. The CATCH project—A field study of air traffic control in Copenhagen (No. CTI Working Paper no. 57): Technical University of Denmark.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: Defining Customer Centred System*. San Francisco, CA: Morgan Kaufman.

Billings, C. E. 1995. Situation awareness measurement and analysis: A commentary. Paper presented at the Proceedings of the International Conference on Experimental Analysis and Measurement of Situation Awareness, Daytona Beach, FL.

Billings, C. E. 1997. *Aviation Automation: The Search for a Human-Centered Approach*. Mahwah, NJ: Lawrence Erlbaum.

Breznitz, S. 1984. *Cry Wolf: The Psychology of False Alarms*. Mahwah, NJ: Lawrence Erlbaum.

Brudnicki, D. J., and A. L. McFarland. 1997. *User Request Evaluation Tool (URET) Conflict Probe Performance and Benefits Assessment.* (No. MP97W0000112). McLean, VA: MITRE.

Bryan, L. A., J. W. Stonecipher, and K. Aron. 1954. 180-degree turn experiment. *Univ Ill Bull* 54(11):1–52.

Chapanis, A. 1959. *Research Techniques in Human Engineering*. Baltimore, MD: The Johns Hopkins Press.

Civil Aeronautics Board. 1953. Civil Air Regulations Part 4b, 4b.611 (b) C.F.R.

Civil Aeronautics Board. 1957. Civil Air Regulations Part 4b, Amendment 4b-7 C.F.R.

Cohen, E., and R. L. Follert. 1970. Accuracy of interpolation between scale graduations. *Hum Factors* 31(5):481–3.

Cole, E. L., J. L. Milton, and B. B. McIntosh. 1954. *Routine Maneuvers Under Day and Night Conditions, Using An Experimental Panel, Eye Fixations of Aircraft Pilots ix* (No. Technical Report 53-220). Wright-Patterson AFB, OH: USAF WADC.

Conrad R. 1951. Speed and load stress in sensory-motor skill. *Br J Ind Med* 8:1–7.

Cooper, G. E. 1977. *A Survey of the Status of and Philosophies Relating to Cockpit Warning Systems* (No. NASA-CR-152071). Moffett Field, CA: NASA Ames Research Center.

Cotton, B. 2003. For spacious skies—the twenty-first century promise of free flight. Paper presented at the AIAA International Air and Space Symposium and Exhibition: The Next 100 Years, Dayton, OH.

Craik, K. 1943. *The Nature of Explanation*. Cambridge, UK: Cambridge University Press.

Crane, P. M. 1992. *Theories of Expertise as Models for Understanding Situation Awareness (No. ADP006943)*. Williams, AZ: The Armstrong Lab.

Dashevsky, S. G. 1964. Check-reading accuracy as a function of pointer alignment, patterning, and viewing angle. *J Appl Psychol* 48 344–47.

Davenport, W. W. 1978. *Gyro! The Life and Times of Lawrence Sperry*. New York: Charles Scribner.

Dearden, A., M. Harrison, and P. Wright. 1998. Allocation of function: scenarios, context and the economics of effort. *Int J Hum Comput Stud* 52(2):289–318.

de Brito, G. 2002. Towards a model for the study of written procedure following in dynamic environments. *Reliab Eng Saf Sci* 75:233–44.

Degani, A., and E. Weiner. 1990. *Human Factors of Flight-Deck Checklists: The Normal Checklist*. (No. 177549). Moffett Field, CA: NASA Ames Research Center.

Degani, A., and E. Weiner. 1993. Cockpit checklists: Concepts, design, and use. *Hum Factors* 35(2):28–43.

Degani, A., and E. L. Weiner. 1997. Procedures in complex systems: The airline cockpit. *IEEE Trans Syst Man Cybern* 27:302–12.

Dien, Y. 1998. Safety and application of procedures, or how do they have to use operating procedures in nuclear power plants? *Saf Sci* 29:179–87.

Djokic, J., B. Lorenz, and H. Fricke. 2010. Air traffic control complexity as workload driver. *Transp Res Part C Emerg Technol* 18(6):930–6.

Dorris, A. L., T. Connolly, T. L. Sadosky, and M. Burroughs. 1977. More information or more data? Some experimental findings. Paper presented at the Human Factors Society 21st Annual Meeting, Santa Monica, CA.

Elkin, E. H. 1959. *Effect of Scale Shape, Exposure Time and Display Complexity on Scale Reading Efficiency* (No. TR 58-472). Wright-Patterson AFB, OH: USAF WADC.

Endsley, M. R. 1988. Design and evaluation for situation awareness enhancement. Paper presented at the Human Factors Society 32nd Annual Meeting, Santa Monica, CA.

Endsley, M. R., and D. G. Garland. 2000. Pilot situation awareness training in general aviation. Paper presented at the 44th Annual Meeting of the Human Factors and Ergonomics Society, Santa Monica, CA.

Endsley, M. R., and M. D. Rodgers. 1996. Attention distribution and situation awareness in air traffic control. Paper presented at the 40th Annual Meeting of the Human Factors and Ergonomics Society, Philadelphia, PA.

Eriksen, C. W., and B. A. Eriksen. 1974. Effects of noise letter upon the identification of a target letter in a nonsearch task. *Percept Psychophys* 16:143–9.

Erzberger, H. 2004a. *Transforming the NAS: The Next Generation Air Traffic Control System* (No. TP-2004-212828). Moffett Field, CA: NASA Ames Research Center.

Erzberger, H. 2004b. Transforming the NAS: The next generation air traffic control system. Paper presented at the 24th International Congress of the Aeronautical Sciences, Yokohama, Japan.

Erzberger, H., and W. Nedell. 1989. *Design of Automated System for Management of Arrival Traffic*. Moffett Field, CA: NASA Ames Research Center.

Erzberger, H., and R. A. Paielli. 2002. Concept for next generation air traffic control system. *Air Traffic Control Q* 10(4):355–78.

Farber, E., and M. Paley. 1993. Using freeway traffic data to estimate the effectiveness of rear-end collision countermeasures. Paper presented at the Third Annual IVHS America Meeting, Washington, DC.

Ferrell, W. R., and T. B. Sheridan. 1967. Supervisory control of remote manipulation. *IEEE Spectr* 4(10):81–8.

Fitts, P. 1951a. Engineering psychology in equipment design. In *Handbook of Experimental Psychology*, ed. S. S. Sevens, 1287–340. New York: Wiley.

Fitts, P. 1951b. *Human Engineering for an Effective Air-Navigation and Traffic-Control System*. Washington, DC: National Research Council Committee on Aviation Psychology.

Fitts, P., R. E. Jones, and J. L. Milton. 1950. Eye movements of aircraft pilots during instrument-landing approaches. *Aeronaut Eng Rev* 9(2):1–6.

Flach, J. M. 1995. Situation awareness: Proceed with caution. *Hum Factors* 37:149–57.

Fogel, L. 1959. A new concept: The kinalog display system. *Hum Factors* 1(1):30–37.

Fracker, M. L. 1991. *Measures of Situation Awareness: Review and Future Directions* (No. AL-TR-1991-0128). Wright-Patterson AFB, OH: USAF Armstrong Laboratory.

Francis, G. 2000. Designing multifunction displays: An optimization approach. *Int J Cogn Ergon* 4(2):107–24.

Gentner, D., and A. L. Stevens, eds. 1983. *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum.

Gerovitch, S. 2003. Interview with Sheridan. Retrieved November 4 2011, from http://web.mit.edu/slava/space/interview/interview-sheridan.htm.

Gilbert, G. A. 1973. Historical development of the air traffic control system. *IEEE Trans Commun* 21:364–75.

Gosling, G. D. 2002. Analysis of factors affecting the occurrence and severity of air traffic control operational errors. Paper presented at the 2002 Transportation Research Board Annual Meeting, Washington, DC.

Graeber, R. C., and M. M. Moodi. 1998. Understanding flight crew adherence to procedures: The Procedural Event Analysis Tool (PEAT). Paper presented at the Flight Safety Foundation/International Federation of Airworthiness International Air Safety Seminar, Cape Town, South Africa.

Green, P. 1984. *Driver Understanding of Fuel and Engine Gauges* (No. Technical Paper Series 840314). Warrendale, PA: Society of Automotive Engineers.

Hacker, E. W. 1984. Learning from the Past: A Fighter Pilot's Obligation. http://www.globalsecurity.org/military/library/report/1984/HEW.htm (accessed July 29, 2005).

Hanson, R. H., D. G. Payne, R. J. Shively, and B. H. Kantowitz. 1981. Process control simulation research in monitoring analog and digital displays. Paper presented at the The Human Factors Society 25th Annual Meeting, Rochester, NY.

Harper, R., and J. Hughes. 1991. What a F-ing system! Send 'em all to the same place and then expect us to stop 'em hitting: Making technology work in air traffic control. In *Technology in Working Order: Studies of Work, Interaction, and Technology*, ed. G. Burton, 127–44. Cambridge, UK: Rank Xerox Cambridge EuroPARC.

Heglin, H. J. 1973. *NAVSHIPS Display Illumination Design Guide: Human Factors* (No. NELC/TD223). San Diego, CA: Naval Electronics Laboratory Center.

Hersey, M. D. 1923. *Aeronautic Instruments. Section I: General Classification of Instruments and Problems Including Bibliography*. (No. 125). Langley, VA: NASA.

Hodge, M. H., and S. R. Reid. 1971. The influence of similarity between relevant and irrelevant information upon a complex identification task. *Percept Psychophys* 10:193–6.

Hughes, J. A., D. Randall, and D. Shapiro. 1992. Faltering from ethnography to design. Paper presented at the ACM Conference on Computer Supported Cooperative Work, Toronto, ON.

Iwasaki, S. 1996. Speeded digit identification under impaired perceptual awareness. Paper presented at the Toward a Science of Consciousness 1996 Conference, Tucson, AZ.

Johnson-Laird, P. N. 1983. *Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness*. Cambridge: Harvard University Press.

Jones, D. G., and M. R. Endsley. 2000. Can real-time probes provide a valid measure of situation awareness? Paper presented at the Human Performance, Situation Awareness and Automation: User Centered Design for the New Millennium Conference, Savannah, GA.

Kerns, K. 1991. Data-link communications between controllers and pilots: A review and synthesis of the simulation literature. *Int J Aviat Psychol* 1:181–204.

Knorr, D. 2003. *Free Flight Program Performance Metrics to Date: December 2003*. Washington, DC: Federal Aviation Administration.

Kopardekar, P., K. Bilimoria, and B. Sridhar. 2007. Airspace configuration concepts for the next generation air transportation system. *Air Traffic Control Q* 16:313–36.

Kuchar, J. K. 1996. Methodology for alerting-system performance evaluation. *J Guid Control Dyn* 19:438–44.

Kuchar, J. K., and A. Drumm. 2007. The traffic alert and collision avoidance system. *Lincoln Lab J* 16:277–96.

Lagu, A., and S. J. Landry. 2011. Roadmap for the next generation of dynamic function allocation theories and strategies. *Hum Factors Ergon Manuf* 21(1):14–28.

Landry, S., T. Farley, and T. Hoang. 2005. Expanding the use of time-based metering: Multi-Center Traffic Management Advisor. Paper presented at the 6th USA/Europe ATM 2005 R&D Seminar, Baltimore, MD.

Latorella, K. A. 1998. Effects of modality on interrupted flight deck performance: Implications for data link. Paper presented at the Human Factors and Ergonomics Society 42nd Annual Meeting, Santa Monica, CA.

Lee, K. K., and T. J. Davis. 1996. The development of the final approach spacing tool (FAST): A cooperative controller-engineer design approach. *J Control Eng Pract* 4:1161–68.

Lee, K. K., and B. D. Sanford. 1998. *Human factors Assessment: The Passive Final Approach Spacing Tool (pFAST) Operationa Evaluation*. Moffett Field, CA: NASA Ames Research Center.

Levy, J. L., D. C. Foyle, and R. S. McCann. 1998. Performance benefits with scene-linked HUD symbology: An attentional phenomenon? Paper presented at the The 42nd Annual Meeting of the Human Factors and Ergonomics Society, Santa Monica, CA.

Ljundberg, M., and A. Lucas. 1992. *The OASIS Air Traffic Management System (No. 28)*. Melbourne, Australia: Australian Artificial Intelligence Institute.

MacKay, W. E. 1999. Is paper safer? The role of paper flight strips in air traffic control. *ACM Trans Comput Hum Interact* 6:311–40.

McGreevy, M. W., and S. R. Ellis. 1986. The effect of perspective geometry on judged direction in spatial information instruments. *Hum Factors* 28:439–56.

McRuer, D., and D. Graham. 1965. Human pilot dynamics in compensatory systems (No. AAFFDL-TR-65-15): USAF.

Mendel, M. B., and T. B. Sheridan. 1986. *Optimal Combination of Information from Multiple Sources*. Arlington, VA: Office of Naval Research.

Midkiff, A., and R. J. Hansman. 1993. Identification of important 'party line' information elements and implications for situational awareness in the data-link environment. *Air Traffic Control Q* 1(1):5–30.

Miller, C. A., and R. Parasuraman. 2007. Designing for flexible interaction between humans and automation: Delegation interfaces for supervisory control. *Hum Factors* 49:57–75.

Monty, R. A. 1973. Keeping track of sequential events: Implications for the design of displays. *Ergonomics* 16:443–54.

Mountford, S. J., and B. Somberg, B. 1981. Potential uses of two types of stereographic display systems in the airborne fire control environment. In *Proceedings of the Human Factors Society*, vol. 25, 235–9. Santa Monica, CA: The Human Factors and Ergonomics Society.

Murphy, M. R., L. A. McGee, E. A. Paler, C. H. Paulk, and T. E. Wempe. 1978. Simulator evaluation of three situation and guidance displays for V/STOL aircraft zero-zero landing approaches. *IEEE Trans Syst Man Cybern* 8:18–29.

NavCanada. 2003. SASS: Sequencing and Scheduling System. Retreived November 4, 2011, from http://www.navcanada.ca/contentdefinitionsfiles/TechnologySolutions/products/StandAlone/sass/SASSen.pdf.

Norman, D. A. 1993. *Things That Make Us Smart*. Reading, MA: Addison-Wesley.

Ockerman, J. J., and A. R. Pritchett. 2004. Improving performance on procedural tasks through presentation of locational procedure context: An empirical evaluation. *Behav Inf Technol* 23(1):11–20.

Olmos, O., C. D. Wickens, and A. Chudy. 1997. Tactical displays for combat awareness: An examination of dimensionality and frame of reference concepts, and the application of cognitive engineering. Paper presented at the 9th International Symposium on Aviation Psychology, Columbus, OH.

Parasuraman, R., and M. Byrne. 2003. Automation and human performance in aviation. In *Principles of Aviation Psychology*, ed. P. Tsang and M. Vidulich, 311–56. Mahwah, NJ: Lawrence Erlbaum.

Parasuraman, R., M. Mouloua, R. Molloy, and B. Hilburn. 1993. Adaptive function allocation reduces performance costs of static automation. Paper presented at the Proceedings of the Seventh International Symposium on Aviation Psychology, Columbus, OH.

Parasuraman, R., and V. Riley. 1997. Humans and automation: Use, misuse, disuse, abuse. *Hum Factors* 39:230–53.

Parasuraman, R., T. B. Sheridan, and C. D. Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Trans Syst Man Cybern* 30:286–97.

Parker, D., and R. Lawton. 2000. Judging the use of clinical protocol by fellow professionals. *Soc Sci* 51:669–677.

Prevot, T., S. Sheldon, J. Mercer, P. Kopardekar, E. Palmer, and V. Battiste. 2003. ATM concept integrating trajectory-orientation and airborne separation assistance in the presence of time-based traffic flow management. Paper presented at the 22nd Digital Avionics System Conference, Indianapolis, IN.

Pritchett, A. R. 2001. Reviewing the role of cockpit alerting systems. *Hum Factors Aerosp Saf* 1(1):5–38.

Pritchett, A. R., R. J. Hansman, and E. N. Johnson. 1995. Use of testable responses for performance-based measurement of situation awareness. In *Experimental Analysis and Measurement of Situation Awareness*, ed. M. R. Endsley and D. J. Garland, 201–10. Daytona Beach, FL: Embry-Riddle Press.

Pulat, B. M., and M. A. Ayoub. 1979. A computer-aided instrument panel design procedure. *Proc Hum Factors Soc* 23:191–92.

Pylyshyn, Z. 2003. Mental imagery: In search of a theory. *Trends Cogn Sci* 7(3):113–18.

Rani, P., J. Sims, R. Brackin, and N. Sarkar. 2002. Online stress detection using psychophysiological signals for implicit human-robot cooperation. *Robotica* 20:673–85.

Regal, D. M., W. H. Rogers, and G. P. Boucek. 1988. *Situational Awareness in the Commercial Flight Deck: Definition, Measurement, and Enhancement* (No. Technical Paper 881508). Warrendale, PA: Society of Automotive Engineers.

Robinson III, J. E., T. J. Davis, and D. R. Isaacson. 1997. Fuzzy reasoning-based sequencing of arrival aircraft in the terminal area. Paper presented at the AIAA Guidance, Navigation, and Control Conference, New Orleans, LA.

Roscoe, S. N. 1968. Airborne displays for flight and navigation. *Hum Factors* 10(4):321–32.

Roscoe, S. N. 1997. Horizon control reversals and the graveyard spiral. *CSERIAC Gatewa* 7(3):1–4.

Roscoe, S. N., L. Corl, and R. S. Jensen. 1981. Flight display dynamics revisited. *Hum Factors* 23(3):341–53.

Sabeh, R., W. R. Jorve, and J. M. Vanderplas. 1958. *Shape Coding of Aircraft Instrument Zone Markings* (No. Technical Note 57-260). Wright-Patterson AFB: USAF WADC.

Sanders, and E. J. McCormick. 1993. *Human Factors in Engineering and Design*. New York: McGraw-Hill.

Sarter, N. B., and D. D. Woods. 1991. Situation awareness: A critical but ill-defined phenomenon. *Int J Aviat Psychol* 1(1):45–57.

Scallen, S. F., P. A. Hancock, and J. A. Dudley. 1996. Pilot performance and preference for short cycles of automation in adaptive function allocation. *Appl Ergon* 26(6):397–403.

Schmidt, K., and L. Bannon. 1992. Taking CSCW seriously: Supporting articulation work. *Comput Support Coop Work* 1(1):7–40.

Schneiderman, B. 1998. *Designing the User Interface*. Cambridge, MA: Addison-Wesley.

Seidler, K., and C. D. Wickens. 1992. Distance and organization in multifunction displays. *Hum Factors* 34:555–69.

Selcon, S. J., and R. M. Taylor. 1989. Evaluation of the situational awareness rating technique (SART) as a tool for aircrew systems design. Paper presented at the NATO AGARD conference on situational awareness in aerospace operations, Springfield, VA.

Sheridan, T. B. 1992. *Telerobotics, Automation, and Human Supervisory Control*. Cambridge: The MIT Press.

Sheridan, T. B. 1998. Allocating functions rationally between humans and machines. *Ergon Des* 6(3):20–5.

Simmonds, G. R., M. Galer, and A. Baines. 1981. *Ergonomics of Electronic Displays*. Warrendale, PA: Society of Automotive Engineers.

Smith, K., and P. A. Hancock. 1995. Situation awareness is adaptive, externally directed consciousness. *Hum Factors* 37(1):137–48.

Sokkappa, B. G. 1989. *Impact of Metering Methods on Airport Throughput*. (Report No. MP-89W000222) (No. MP-89W000222). McLean, VA: MITRE Corporation.

Sridhar, B., S. R. Grabbe, K. Sheth, and K. Bilimoria. 2007. Initial study of tube networks for flexible airspace utilization. Paper presented at the Guidance, Navigation, and Control Conference 2007, Hilton Head, SC.

Starr, S. L., and J. R. Griesemer. 1989. Institutional ecology, "translations," and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology. *Soc Stud Sci* 19:387–420.

Suchman, L. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge, UK: Cambridge University Press.

Swenson, H., T. Hoang, S. Engelland, D. Vincent, T. Sanders, B. Sanford et al. 1997. Design and Operational Evaluation of the Traffic Management Advisor at the Fort Worth Air Route Traffic Control Center. Paper presented at the 1st USA/Europe Air Traffic Management R&D Seminar, Saclay, France.

Taylor, R. M. 1989. Situational awareness rating technique (SART): The development of a tool for aircrew systems design. Paper presented at the NATO AGARD conference on situational awareness in aerospace operations, Springfield, VA.

Tipper, S. P., and J. Driver. 1988. Negative priming between pictures and words in a selective attention task: Evidence for semantic processing of ignored stimuli. *Mem Cogn* 16(1):64–70.

Vandevenne, H. F., and J. W. Andrews. 1993. Effects of metering precision and terminal controllability on runway throughput. *Air Traffic Control Q* 1:277–97.

VanRullen, and C. Koch. 2003. Competition and selection during visual processing of natural scenes and objects. *J Vis* 3(1):75–85.

VanRullen, and S. J. Thorpe. 2001. The time course of visual processing: From early perception to decision-making. *J Cogn Neurosci* 13(4):454–61.

Veitengruber, J. E. 1977. Design criteria for aircraft warning, caution, and advisory alerting systems. *J Aircr* 15(9):574–81.

Ververs, M. P., and C. D. Wickens. 1998. Head-up displays: Effects of clutter, symbology, intensity, and display location on pilot performance. *Int J Aviat Psychol* 8:377–403.

Vicente, K. J., and J. Rasmussen. 1992. Ecological interface design: Theoretical foundations. *IEEE Trans Syst Man Cyberne* SMC-22:589–606.

Vidulich, M. A. 1992. Measuring situation awareness. Paper presented at the The Human Factors Society 36th Annual Meeting, Atlanta, GA.

Wegerbauer, C. 2005. The Access 5 project—Enabling direct route UAS operations in the U.S. National Airspace System. *Unmanned Veh Mag* 10:3.

Well, A. 1971. The influence of irrelevant information on speeded classification tasks. *Percept Psychophys* 13:79–84.

Whitehurst, H. O. 1982. Screening designs used to estimate the relative effects of display factors on dial reading. *Hum Factors* 24(3):301–10.

Whitfield, D., and A. Jackson. 1983. The air traffic controller's picture as an example of a mental model. Paper presented at the IFAC Conference on Analysis, Design, and Evaluation of Man-Machine Systems, Baden-Baden, Germany.

Wickens, C. D. 2003. Aviation displays. In *Principles and Practice of Aviation Psychology*, ed. P. Tsang and M. Vidulich, 147–199. Mahwah, NJ: Lawrence Erlbaum.

Wickens, C. D., and C. M. Carswell. 1995. The proximity compatibility principle: Its psychological foundation and its relevance to display design. *Hum Factors* 37(3):473–94.

Wickens, C. D., S. Fadden, D. Merwin, and P. M. Ververs. 1998. Cognitive factors in aviation display design. Paper presented at the 17th Digital Avionics Systems Conference, Bellevue, WA.

Wickens, C. D., A. S. Mavor, R. Parasuraman, and J. P. McGee. 1998. The Future of Air Traffic Control: Human Operators and Automation. Washington, DC: National Academy Press.

Wickens, C. D., S. Miller, and M. Tham. 1996. The implications of data-link for representing pilot request information on 2D and 3D air traffic control displays. *Int J Ind Ergon* 18:283–93.

Wickens, C. D., and T. Prevett. 1995. Exploring the dimensions of egocentricity in aircraft navigation displays. *J Exp Psychol Appl* 1(2):110–35.

Wickens, C. D., S. Rice, D. Keller, S. Hutchins, J. Hughes, and K. Clayton. 2009. False alerts in air traffic control conflict alerting system: Is there a "cry wolf" effect? *Hum Factors* 51:446–62.

Wiener, E. L., and R. E. Curry. 1980. Flight-deck automation: Promises and problems. *Ergonomics* 23(10):995–1011.

Wilson, G. F., and C. A. Russell. 2007. Performance enhancement in an uninhabited air vehicle task using psychophysiologically determined adaptive aiding. *Hum Factors* 49:1005–18.

Wittgenstein, L. 1922. *Tractatus Logico-Philosophicus*. London: Routledge & Kegan Paul.

Woods, D. D. 1984. Visual momentum: A concept to improve the cognitive coupling of person and computer. *Int J Man Mach Stud* 21:229–44.

Wright, P., A. Dearden, and B. Fields. 1998. Function allocation: A perspective from studies of work practice. *Int J Hum Comput Stud* 52(2):335–55.

Zhang, J., and D. A. Norman. 1994. Representations in distributed cognitive tasks. *Cogn Sci* 18:87–122.

34 User-Centered Design in Games

Randy J. Pagulayan, Kevin Keeker, Thomas Fuller, Dennis Wixon, Ramon L. Romero, and Daniel V. Gunn

CONTENTS

34.1 INTRODUCTION

The application of user-centered design (UCD) methodologies has had a tremendous impact on the aviation, medical, and technology industries over the past several decades (Preece et al. 1996; Salas and Maurino 2010; Wickens and Hollands 2000). UCD principles and evaluation methods have successfully made our interaction with tools faster, safer, and more efficient. Yet, there has been growing interest in the adaptation of these techniques to make products more pleasurable (Jordan 2000). Instead of focusing on making tools faster, safer, and more efficient, greater emphasis has been placed on making them enjoyable to use (Norman 2005). For more than a decade at Microsoft, we have been applying, refining, and inventing new UCD techniques to improve not only the *usability* of our games, but, more importantly, the *enjoyability*. The purpose of this chapter is to review some principles and challenges in applying UCD techniques to game improvement and evaluation. First, we discuss why games are important followed by a differentiation between games and productivity software. Next, we discuss the principles and challenges that are unique to the design of games. That discussion provides a framework to illustrate the core variables we think one should measure to aid in game design testing and evaluation. Finally, the chapter will conclude with some examples of how those variables have been operationalized in methods used by Studios User Research at Microsoft Studios.

34.2 WHY GAMES ARE IMPORTANT

In the 1950s, computers imposed a technological and economical barrier to entry that was not easy to overcome for the general population. Only scientists, engineers, and highly technical persons with access to universities or large companies were able to use these machines (Preece et al. 1994). As computers became less expensive, more advanced, and more reliable, the technology that was once available to only a small subset of the population began proliferating throughout society and became integral to everyday life. In order to make this transition possible, well-designed interfaces between the user and the technology were required. Video games come from similar origins and appear to be heading down a similar path. Some early attempts at making commercial video games failed due to unnecessarily high levels of complexity. As Nolan Bushnell (cofounder of Atari) states, “No one wants to read an encyclopedia to play a game” (Kent 2000, p. 28). In retrospect, some of the most successful video games were the ones that were indeed very simple. The more recent adoption of “broadening interfaces” such as Nintendo’s Wiimote, Microsoft’s Kinect, or Sony’s Move, as well as the large number of people buying games, shows a similar proliferation of games in society. This too is only possible with well-designed interfaces between the user and technology.

34.2.1 Games Industry Generates Enormous Revenue

Within the first 24 hours of its release, the video game *Halo 3* generated approximately $170 million dollars in sales, in the United States alone, rivaling the film industry's opening day record for *Spiderman 3* as well as the book industry's opening day record for *Harry Potter and the Deathly Hallows* (Chalk 2007; Microsoft 2007). Video games are one of the fastest growing forms of entertainment to date, and today the video games industry generates enormous revenue. According to the Entertainment Software Association (ESA), revenue from computer and console games has grown from 2.6 billion dollars in 1996 to 11.7 billion dollars in the year 2008 (ESA 2009). Approximately 298 million games were sold in 2008 and approximately 43% of Americans purchased or planned to purchase one or more games in 2009 (ESA 2009). These statistics do not even consider the international importance of video games. To put this into perspective, the combined U.S./Canada movie box office for 2008 was $9.6 billion (MPAA 2009), which is approximately $2 billion less than the revenue from computer and console games from the United States alone.

34.2.2 Games Are Not Just for "Gamers" Anymore

The revenue created by the games industry, one that rivals the U.S. film industry, suggests that gaming appeals to more than just a niche market. In fact, video games are no longer associated with the stereotypical image of a gamer as an adolescent male; women aged 18 and older represent a larger portion of the game playing population (34%) compared to boys aged 17 and younger (18%), and the average age of those who play video games is 35 (ESA 2009). In addition, gaming has become increasingly more popular with the older population with 25% of all gamers being over the age of 50 (ESA 2009). All together, a majority of American households play computer or video games and with game revenue rivaling the motion picture revenue, it is no wonder that the people who play games are as diverse as the games they play.

34.2.3 Games Push Technology and Research

In addition to driving revenue, video games have also helped to shape and drive technology and research. For example, during the early 1980s, IBM commissioned Sierra Online to develop a game to show off the advanced graphics capabilities and three channel sound of their upcoming IBM PCjr computers. While the PCjrs didn't fare well in the marketplace, Sierra's creation actually might have helped IBM's competition. The resulting game, *King's Quest*, arguably helped drive sales of the PCjr compatible Tandy 1000 computer by showcasing its enhanced graphics and sound capabilities. A more recent example of games driving technology can be seen in the Xbox 360 Xenon processor developed by IBM and the Sony cell processor developed by Sony, Toshiba, and IBM. These processors were purposefully built for the Xbox 360 and Playstation 3, required several years of design, research, and iteration in order to build, and pushed the then current microprocessor technology to its limits—all for the primary goal of providing these consoles with the computing power required to support next generation games (Shippy and Phipps 2009).

34.2.4 Games Are a Source of Interface Innovation

Productivity applications typically strive for interface consistency from one version to the next. In that market segment, learning a new interface is considered a cost by business and a burden by users. Changing interfaces should only be done if the gains in efficiency outweigh the costs. Indeed, even the input devices associated with productivity applications have been relatively consistent for a long time. Yet, while the mouse and keyboard have stayed essentially the same for 30 years, game controllers evolve and many new games use new and novel input devices (e.g., guitars, game mats, Nintendo Entertainment System power gloves, Microsoft Kinect). In fact, gamers demand novelty, and, consequently, these evolving and novel devices create game design and ergonomic challenges that are very important to study. Moreover, each interface innovation developed for a game can potentially touch millions of people. This creates a chance to extend the interface innovation out of the games market and into other venues. For this reason, games can and should be viewed as an experimental interface testing ground.

34.3 GAMES VERSUS PRODUCTIVITY APPLICATIONS

As the games industry matures, games become more complex and varied making them difficult to narrowly define. Articulating a clear and succinct set of principles that capture the essence of games is not straightforward. Yet, shedding some light on the difference between games and productivity applications might make it easier to see how traditional UCD techniques must be adapted in order to have the most impact on games. When comparing games to productivity software, there are principles and methods that can be successfully applied to both. Regardless of the domain, the same techniques would be applied to understand a misleading button label or a confusing process model. At the same time, there are fundamental differences between productivity applications and gaming applications. Some of the differences can be fairly clear; games are usually more colorful, wacky, and escapist than productivity applications, with the inclusion of interesting story lines, or animated characters. Others may not be. In productivity applications, unclear labeling in the user interface (UI) is affecting my *productivity*; in games, the unclear labeling in the UI is affecting my *fun*.

UCD principles have not reached game makers to the degree that they have influenced other electronic applications.

Sections 34.3.1 to 34.3.4 will describe some differences that have important implications for user research on games.

34.3.1 The Goals Are Different

At a very basic level, the design goals, as well as user goals, are often quite different between games and productivity applications. If approached from a tools perspective, productivity applications are tools that help consumers be productive whereas games help consumers have fun. This perspective is useful because it allows us (1) to prioritize usability and enjoyability issues based on their impact on fun; (2) to see why the role of designer intent is so important in usability/enjoyability testing of games; and (3) to focus on the similarities of UCD principles between tool and game testing to more clearly see how traditional UCD techniques can be adapted to game development.

Because productivity applications are focused on making the user productive, the focus of design and usability is to allow the user to produce an improved work product or result with less effort, higher quality, and greater satisfaction. However, games are intended to be a pleasure to play. Ultimately, games are like movies, literature, and other forms of entertainment. They exist in order to stimulate thinking and feeling. Their outcomes are more experiential than tangible. This is not to say that word processors or other tools cannot be a pleasure to use, or that people do not think or feel when using them, but pleasure or other feelings are rarely their primary design intention. In a good game, both the outcomes and the journeys are rewarding. This fundamental difference leads us to devote more of our effort to measuring perceptions while productivity applications focus more on task completion.

34.3.2 The Role of Designer Intent

The general design goal for productivity applications is to *enable users to be productive*; for games, the goal is to *enable users to have fun*. For testing purposes, it is generally easier for usability practitioners to spot user deviations away from designer intent within productivity applications than it is in games. Most people can easily see when productivity is blocked; determining if, and when, fun is blocked is a bit more elusive. For example, it is unlikely that the designer intends for users to take four attempts to print their work so that it does not appear cropped in a productivity application. If the blockers to more efficient performance could be identified and removed, and users could successfully print on average with just one attempt, the redesigned application would make the users more productive and map better onto the designer intent. However, in games, four attempts at beating a boss might map perfectly onto the designer's intent. In fact, defeating a boss with only one attempt might actually be a blocker to fun if it is perceived as too easy. Six attempts might make the game too hard, but four attempts might be "just right." Moreover, four attempts might be just right for *some* users and not others. Thus, for games, we must rely more heavily on mapping user behavior against explicit designer intent. The designer holds the vision for their creation as well as the vision for what makes their game fun. User research on games requires close collaboration with design in order to spot deviations away from intent that could impact fun.

34.3.3 Games Must Challenge

Perhaps the most important distinction between games and productivity applications is the complex relationship between challenge, consistency, and frustration. Productivity applications, and usability testing efforts associated with them, strive to remove blockers to productivity—challenging flows, difficult navigation, unclear labeling, and so on. With games, we strive to remove blockers to fun—cumbersome weapon changing, uncomfortable controller mappings, confusing gameplay, and so on. Yet, certain kinds of gameplay challenges and difficulty don't block fun but are intentionally placed there by the designer to enhance it; they add to the *enjoyability*. To illustrate this difference, consider the following example. It has been said that the easiest game to use would consist of one button labeled "Push." When you push it, the display says "YOU WIN." This game would have few, if any usability issues; but it would not be fun either. Indeed, it has enjoyability issues galore: lack of challenge, lack of strategy required, lack of gameplay trade-offs, lack of interesting goals, and lack of replayability (to name just a few) all conspire to make this a pretty stale game. With user research in games, testing efforts should focus more on mitigating those factors that most impact enjoyability and impede fun. One should ensure that challenge adds to enjoyability and doesn't detract from it and that it is part of an intentional gameplay element and not a usability issue.

34.3.4 Industry Competition Is Intense

Competition within the games industry is more intense than other software domains. Games compete with each other as well as many other forms of entertainment for your attention. Pretend that you have to write a book chapter. For that task, there are a limited number of viable tools: hand writing, dictation, a typewriter, or a limited set of software applications. But, imagine that you've got a few hours to kill and you want to avoid productivity at all costs. You could read the biography of Nikola Tesla, argue politics with your best friend, watch a mystery film, or race your Ferrari 458 Italia 160 mph down Nürburgring in *Forza Motorsport 4* (Microsoft 2011). This element of choice, combined with the large number of games, means that many games will fail (just as many films, and TV shows fail). In contrast, a productivity application that is the "standard" will enjoy massive and sustained success.

Many of the characteristics mentioned above are not unique to games. But all of these characteristics are relevant to the discussion of the ways in which user research must adapt to be useful for game development. Later, we will discuss how these differences create particular challenges for both the design and evaluation of games.

34.4 TYPES OF GAMES

Below we review some common gaming platforms and game types.

34.4.1 Platforms

One of the simplest classifications of games is by the platform or hardware that they are played on. Cassell and Jenkins (2000) differentiates games played on a PC from those played on a console. Crawford (1982) divides games into even finer categories: arcade/coin-op, console, PC, mainframe, and handheld. Different gaming platforms can be differentiated by their technical capabilities, physical control interface, visual interface, and the context in which they are played.

34.4.1.1 Personal Computer

A useful distinction can be made between PC games that are normally acquired through retail outlets on a CD/DVD, persistent world games in which much of the content "lives" on the Internet, and casual, web-based games. While there are many technical differences between these kinds of games, the important distinction is the business model and level of investment required by the players. Most retail games need to rely heavily on flash, reputation, and recognizable novelty to attract users to a relatively large investment. Continued investment in the game is only useful to build a reputation or to convince customers with higher thresholds that they will get their money's worth from the game. But massively multiplayer online (MMO) persistent worlds require continued user investment to obtain monthly subscription fees. The long learning curves and reinforcement schedules in MMO are tailored to make players invest more time, money, and effort for long-term rewards. Many casual gamers are attracted to free, familiar games. For these casual users, game play is squeezed in between activities that are more important to them. As a result, games with very little learning investment are very appealing to casual gamers. Removing penalties for setting aside the game (for minutes, hours, days, or forever) can mean the difference between successful and unsuccessful casual games. See Section 34.4.2.2 for more details.

34.4.1.2 Console

For user research purposes, the most important unique characteristic of console video game systems is the advantage of a fixed set of hardware. By contrast with PC games, there is very little game setup, minimal maintenance efforts (e.g., few software patches, video card conflicts, and so on), and a consistent set of input device capabilities. Some game genres are more popular on console than PC. This difference in popularity can usually be attributed to differences in the input devices that are typically associated with each platform. It's hard to enter text using a controller. Thus, most popular MMO games are still published for PCs in spite of the fact that fast internet connections and hard drives are becoming more popular additions to console systems. This may change as it becomes easier to attach other devices such as keyboards or voice command hardware to consoles, which would make communication more robust.

More recently, casual games are also available for all console systems, including Xbox 360, where consumers can download, try, and purchase casual games directly from the console via a broadband connection.

34.4.1.3 Portable Devices

First released in 1989, the Game Boy and Game Boy Color (1998) combined to sell over 118 million units, and this success has only accelerated with the Game Boy Advance (2001; 80 million+ units) and Nintendo DS (2004; 125 million+ units) (Business Week 2006; Nintendo 2005). Nintendo succeeded largely by catering to the massive mainstream demand for accessible games; starting in 1989 by selling a whopping 35 million units of *Tetris* and 14 million units of *Super Mario Land*, and later selling more than 34 million *Pokemon*-related cartridges (Nintendo 2005). In fact, the portable gaming market is littered with devices that have tried to beat Nintendo with higher tech devices and more action-oriented games. More recent trends point to an impending explosion in mobile phone gaming. Since 2005, *Tetris*, now 26 years old, has sold to more than 100 million cell phones (Maximejohnson 2010). As mobile phones becomes more accessible and improve the visual display, gaming upon them becomes more and more popular.

Consumers primarily view mobile phones as communication devices, which are not meant to replace the gaming experience available on handhelds such as the Nintendo DS and Sony PSP (Moore and Rutter 2004). Unlike handheld gaming devices, mobile phones are generally carried at all times. This creates an opportunity to provide gaming at anytime, anywhere (Hyman 2005; IDGA 2005). However, with mobility comes the greater need to design for time, location, social, emotional, and motivational context (Sidel and Mayhew 2003).

34.4.2 Game Types

34.4.2.1 Retail Games (National Purchase Diary Classifications)

The National Purchase Diary (NPD) Group (a marketing company) uses a fine grained classification scheme for game type which is referred to quite often in the games industry. They offer the following classes and subclasses, as shown in Table 34.1. Games can also be categorized in a more granular fashion by splitting each genre into a subgenre. For instance, action games can be divided into Action Driving Hybrid (*Crackdown 2, Saints Row 2*, etc.), Action combat (*Tom Clancy's Splinter Cell: Conviction, Tomb Raider: Underworld*, etc.), and Platformer (*Ratchet & Clank Future: A Crack in Time, Super Mario Bros Wii*, etc.). In general, the genres represent types of gameplay mechanics, themes, or content.

TABLE 34.1
The National Purchase Diary Group Super Genre Classification Scheme for Games

Category	Description
Action	Control a character and achieve one or more objectives with that character.
Fighting	Defeat an opponent in virtual physical hand-to-hand or short-range combat.
Racing	Complete a course before others do and/or accumulate more points than others while completing a course.
Shooter	Goal is to defeat enemies in combat with ranged weapons (first-person shooters, third-person shooters).
Strategy	Strategically manage resources to develop and control a complex system in order to defeat an opponent or achieve a goal. The goal is to defeat opponent(s) using a large and sophisticated array of elements.
Role playing	Control a character that is assuming a particular role in order to achieve a goal or mission. Rich story and character development are common.
Family entertainment	The primary objective is to interact with others and/or to solve problems. This genre includes puzzles and parlor games.
Children's entertainment	Same as family entertainment but geared to a younger audience.
Sports	Manage a team or control players to either win a game or develop a team to championship status. Involves individual, team, and extreme sports.
Adventure	Control a character to complete a series of puzzles that are required to achieve a final goal.
Arcade	Games on coin-op arcade machines or games that have similar qualities to classic arcade games. Generally, they are fast paced, action games.
Flight	Plan flights and pilot an aircraft in a realistic, simulated environment.
All other games	Educational, compilations, nonflight simulators, rhythm games, and so on.

34.4.2.2 Casual Games

Casual games is a rapidly growing segment of the video games market, producing an estimated $500 million in online sales in 2005 (IDG 2005). Casual games as a whole are appealing to a broad range of users, can be found on all platforms, and represent several genres. For these reasons, they do not fit into either of the previous categorization schema. Casual games generally meet the following requirements: (1) are easy to start and control; (2) can be enjoyed in small time intervals; (3) do not require an extensive investment in time to enjoy or progress, and therefore usually don't have deep linear storylines; and (4) have discreet goals and rules that are intuitive or easy to learn. Casual games often have a significantly lower development cost, can be distributed digitally (or via physical media), offer trial versions and lower retail prices (or are free), and utilize lower system requirements and storage space. These requirements and attributes make these games well suited for platforms such as PC distributed via the web and mobile devices, where they are most common.

Websites such as Real Arcade, Pogo.com, Yahoo! Games, and MSN Games provide free online casual games, pay-per-play casual games, subscription services, and downloadable games that include a free trial. The downloaded games can be purchased for $5–$20. Mobile phone carriers and a variety of websites provide casual games for mobile devices. Although there are feature rich, 3D noncasual games available for mobile devices, the top selling titles are casual games such as *Tetris, Bejeweled, Bowling*, and *Pac Man* (Hyman 2005).

34.4.3 Recent Trends in Gaming

34.4.3.1 Physical Gaming and New Input Paradigms

New input paradigms are an ongoing area of exploration and innovation in games that will continue into the future. The most recent innovation has been the advent of physically based at-home video gaming as with the highly popular Wii gaming system (Nintendo 2006). Additionally, in recent years there have been several highly successful games that require the user to simulate musical instrument play via rhythm-based gaming like in *Guitar Hero* (Activision 2005) or *Rock Band* (Electronic Arts 2007) using simulated guitar controllers or, in some cases, to closely mimic actual musical play like when using drums in *Rock Band* or singing in *SingStar* (Sony Computer Entertainment 2004). Historically these video game types date back to at least the 1990s. There were arcade games in the mid-to-late 1990s requiring physical movement to play including *Alpine Racer* (Namco 1995) and *Dance Dance Revolution* (Konami 1998). Karaoke-style singing was popularized in Japan in the 1970s, though the idea of turning it into a game with a scoring system did not arise until the aforementioned *SingStar*. Looking into the future, it appears the trend toward at-home physically based video gaming will only continue as the Playstation Move (Sony Computer Entertainment 2010) and Kinect (Microsoft 2010) have each been recently released and both systems rely on improved gesture and physically based control input systems.

New input paradigms create immense complications when considering user-centered game design because we must now account for differing talent levels of the individual players. In the game *SingStar*, to be successful (from the game design's perspective), a player must have some ability to hear and sing at various pitches in order to succeed in the game. In some cases, the game does mimic exactly the real world activity, but instead is based on an approximation of how to do something in the real world. This is the case with most *Rock Band* and *Guitar Hero* gameplay. In these cases, having actual experience and skills with a guitar may impede a player's ability to adapt to the modified guitar-controller interface. Finally, there are other factors to consider. Similar to the talent requirement for singing, in physically based gaming, we must account for the fatigue level, flexibility, and coordination of potential players—even when considering a relatively simple game such as the archery mini-game in *Wii Sports Resort* (Nintendo 2008).

34.4.3.2 Social Network Gaming

Gaming on social network sites, such as Facebook, is the most popular new trend in internet gaming. *FarmVille* launched on Facebook in June 2009 and had bloomed to over 80 million active users as of April 2010 (Gardner 2009; Facebook 2010).

Earlier social successes, such as *Second Life* (2003), innovated on AOL's chat rooms by giving players a "sand box" like virtual world that they could mold and shape to fit their identities and their group connection needs. Inside these spaces, persistent clubs could form to bring people with common interests together across otherwise insurmountable distances. MMO games such as *World of Warcraft* built economies in their game world to necessitate the formation of friendships, allies and guilds out of strangers.

Social network games borrow from these types of games and extend the social equation in a way that has successfully attracted a mainstream audience. Like *World of Warcraft*, social connections are required to succeed. Players must recruit friends as neighbors and connect with them daily through gift-giving in order to advance. This recruiting process serves a viral marketing channel to bring new players into the game. The extension comes primarily from reducing social barriers. Virtual worlds with strong in-group communities require intense time commitment and expert knowledge. Social networking games are free and can be played in 5 minutes a day, asynchronously, to suit busy, varied schedules, and time zones. They use socially normative themes such as farming, pets, restaurants, and cities as their inspiration, title, and language. Then they use familiar shared events such as holidays and seasons to introduce freshness and novelty on a daily basis. They provide social safety by using a trusted social network (Facebook) for authentication and identity and largely eliminate direct interaction. Virtual gifts are free to the giver and the receiver but still valuable for advancing in the game. Through these social lubricants, social networking games have attracted a massive, mainstream audience. The average player is 43 and female. This typical player might have logged thousands of hours on *Solitaire* and *Minesweeper* but wouldn't have called themselves gamers.

34.4.3.3 Cross Platform

Cross-platform convergence is an emerging trend in the games industry that complicates the classification of games by platform. Developers are now creating multiple versions of a game that interact with one another across platforms. For example, Nintendo games such as *Animal Crossing* (Nintendo of America 2002), *The Legend of Zelda: The Wind Waker* (Nintendo of America 2003), *Harvest Moon* (Nintendo of America 2003) allow users to connect a Gameboy portable device to the GameCube console to access additional features and content, or use the Nintendo DS as a touch screen controller. Another example of cross-platform games would be a title where the player may execute specific planning or strategic tasks on a mobile device at any time, then have those actions affect the full game experience the next time it is played on a console or PC. This cross-platform design effect has been referred to as a reciprocation effect (Yuen 2005).

Cross-platform convergence creates new challenges in user centered game design. The act of switching between different platforms means the game must be adapted to multiple interfaces. Users must be informed how and why they should take advantage of the alternate platform. Users must also be made aware of the alternate platform version's existence so they can choose to take advantage of it. For example, in 2010, Microsoft released a Facebook game called *Toy Soldiers: Match Defense* (Microsoft Game Studios 2010) and shortly thereafter released the Xbox 360 game *Toy Soldiers* (Microsoft Game Studios 2010). Several relationships between the games were established. Playing the Facebook version of the game unlocked or awarded certain features in the Xbox version of the game. Meanwhile, the Xbox version of the game included several hints or loading tips, pointing out that there were benefits to playing the Facebook version. Ensuring success at each point in the seams between the games and platforms represents a new and ongoing responsibility for user-centered game design now and into the future.

The next generation of cross-platform experiences may involve matching players for real-time, or asynchronous, competitive and cooperative multiplayer experiences regardless of what platform they are on (portable, PC, console, etc.). As these trends become realized, interesting challenges and opportunities for game developers will mount as games will have to be designed with the limitations of the platform in mind. Indeed, spanning platforms with a single game will not be as simple as porting it. Careful consideration will have to be made with regard to the unique user needs associated with each platform they are interacting with.

34.5 PRINCIPLES AND CHALLENGES OF GAME DESIGN

Having differentiated games from other applications we can look at some of the unique issues in game design and evaluation.

34.5.1 Identifying the Right Kind of Challenges

Games are supposed to be challenging. This requires a clear understanding of the difference between good challenges and frustrating usability problems. Most productivity tools struggle to find a balance between providing a powerful enough tool set for the expert, and a gradual enough learning curve for the novice. However, no designer consciously chooses to make a productivity tool more difficult. After all, doing so would go against the main design goal of making users productive. The ideal tool enables users to experience challenge only in terms of expressing their own creativity. For games, learning the goals, strategies, and tactics to succeed is part of the fun.

Unfortunately, it is not always clear which tasks should be intuitive (i.e., easy to use) and which ones should be challenging. Input from users becomes necessary to distinguish good challenges from incomprehensible design. Take a driving game for example. It's not fun having difficulty making your car move forward or turn. But learning to drive is still a fundamental part of the challenge in the game. While all cars should use the same basic mechanisms, it may be fun to vary the ways that certain cars respond under certain circumstances. It should be challenging to identify the best car to use on an icy, oval track as opposed to a rally racing track in the middle of the desert. The challenge level in a game must gradually increase in order to maintain the interest of the player.

34.5.2 Addressing Different Skill Levels

Unfortunately, all players don't start from the same place in terms of gaming experience or talent. Obviously, frequent failure can be a turn off. Success that comes too easily can also become repetitive. Games must address the problem of meeting all players with the correct level of challenge. Tuning a game to the right challenge level is called "game balancing."

There are many ways to balance the difficulty of the game. The most obvious way is to let players choose the difficulty themselves. Many games offer the choice of an easy, medium or hard difficulty level. While this seems like a simple solution, it is not simple to identify exactly how easy the easiest level should be. Players want to win, but they do not want to be patronized. Too easy is boring and too hard is unfair. Either experience can make a person cease playing.

Another approach to varying skill levels is to provide explicit instruction that helps all users become skilled in the game. You might imagine a tutorial in which a professional golfer starts by explaining how to hit the ball and ends by giving instruction on how to shoot out of a sand trap onto a difficult putting green. Instruction, however, need not be presented in a tutorial. It could be as direct as automatically selecting the appropriate golf club to use in a particular situation with no input from the user, similar to the notion of an adaptive interface, where the interface provides the "right information" at the "right time."

The environments, characters, and objects in a game provide another possibility for self-regulation. Most games will offer the player some choices regarding their identity, their opponents, and their environment. The better games will provide a variety of choices that allow users to regulate the difficulty of their first experiences. With learning in mind, it is not uncommon for the novice player to choose a champion football team to play against a weak opponent. As long as the player can distinguish the good teams from the bad ones, and the teams are balanced appropriately, users will be able to manage their own challenge level.

Some games take it even further by identifying the skill level of the player and regulating the instruction level appropriately. In this situation, instruction can be tuned to the skill level of the player by associating it with key behavioral indicators that signifies that the player is having difficulty. If the game does not detect a problem, it does not have to waste the player's time with those instructions. In *Halo 2* (Microsoft Game Studios 2004) the game detects difficulties that a player may have with certain tasks. For example, to get into a vehicle, the player must press and hold the "X" button on their controller when standing next to the driver's seat of that vehicle. If the player is standing in the right position, but taps the "X" button repeatedly (instead of holding the button down), the game will present a more explicit instruction to press and *hold* the button. This is just one of many dynamic instructions that appear throughout *Halo 2* based on behavioral indicators of player difficulty.

Productivity tools have implemented similar problem-identification features, but often with mixed success due to the open nature of tasks in most productivity applications. Good game tutorials have succeeded by setting clear goals and completely constraining the environment. Doing so focuses the user on the specific skill and simplifies the detection of problematic behavior. Other lessons from game tutorial design will be described in Sections 34.6.2.3, 34.7.2.2.1, and 34.7.4.2.1 of this chapter.

Another in-game approach to autoregulating the difficulty level requires adjusting the actual challenge level of the opponents during the game. Evaluating the success of the player and adjusting the opponent difficulty during the game is often called "dynamic difficulty adjustment" or "rubber-banding." When players perform very skillfully, their performance is moderated by computer-generated bad luck and enhanced opponent attributes. In a football game, the likelihood of fumbling, throwing an interception, or being sacked, may increase as the player increases their lead over their opponent. Even though this may seem like a good solution, there can be a downside. Most people would prefer to play a competitive game (and win) than to constantly trounce a less-skilled opponent. However, overdeveloped rubber-banding can cheat a skilled player out of the crucial feeling of mastery over the game.

A final approach focuses on providing tools that maximize the ability of the trailing player to catch up with the leading player. The key for the game designer is to think of ways to maintain challenge, reward, and progress for the unskilled player without severely hampering the skilled player.

One interesting and explicit example of this is found in *Diddy Kong Racing* (Nintendo of America 1997). In this game, the racer can collect bananas along the roadway. Each banana increases the top speed of your car. The player can also collect missiles to fire forward at the leading cars. Each time you hit a car with a missile it not only slows the car's progress, but it jars loose several bananas that the trailing player can pick up. Thus, trailing players have tools that they can use to catch the leaders even if the leaders are not making any driving mistakes. The chief distinction between this and dynamic difficulty adjustment is that the game is not modifying skills based on success. Instead, the rules of the game provide the trailing player with known advantages over the leader.

34.5.3 Rewarding Players Appropriately

Explicit or slow reinforcement schedules may cause users to lose motivation and quit playing a game. Because playing a game is voluntary, games need to quickly grab the user's attention and keep them motivated to come back again and again. One way to accomplish this is to reward players for continued play. Theories of positive reinforcement suggest behaviors which lead to positive consequences tend to be repeated. Thus, it makes sense that positive reinforcement can be closely tied to one's motivation to continue playing a game. However, it is less clear which types of reinforcement schedules are most effective.

Although it's not necessarily the model that should be used for all games, research suggests that continuous reinforcement schedules can establish desired behaviors in the quickest amount of time (Domjan 2010; Mazur 2006). Unfortunately, once continuous reinforcement is removed, desired behaviors extinguish very quickly. Use of partial reinforcement schedules take longer to extinguish desired behaviors, but may take too long to capture the interest of gamers. Research suggests that variable ratio schedules are the most effective in sustaining desired behaviors (Jablonsky and DeVries 1972). This kind of schedule is a staple of casino gambling games in which a reward is presented after a variable number of desired responses. Overall, there is no clear answer. Creating a game that establishes immediate and continued motivation to continue playing over long periods of time is a very complex issue.

Another facet of reinforcement systems that may impact enjoyment of a game is whether the player attributes the fact that they have been playing a game to extrinsic or intrinsic motivations. Intrinsic explanations for behavior postulate that the motivators to perform the behavior come from the personal needs and desires of the person performing the behavior. Whereas extrinsically motivated behaviors are those that people perform in order to gain a reward from or please other people. In research on children's self-perceptions and motivations, Lepper, Greene, and Nisbett (1973) discovered that children who were given extrinsic rewards for drawing were less likely to continue drawing than those who had only an intrinsic desire to draw. The conclusion that they drew is that children perceived their motivation to draw as coming from extrinsic sources and thus discounted their self-perception that they liked to draw.

The same may be true of reward systems in games (Lepper and Malone 1987; Malone 1981). To a certain degree, all reinforcement systems in games are extrinsic because they are created or enabled by game developers. But, some reward systems are more obviously extrinsic than others. For instance, imagine the following rewards that could be associated with combat in a fantasy role-playing game (RPG). The player who slays a dragon with the perfect combination of spell casting and swordplay may acquire the golden treasure that the dragon was hoarding. In this situation, the personal satisfaction comes from being powerful enough to win and smart enough to choose the correct tactics. The gold is an extrinsic motivator. The satisfaction is intrinsic. By analogy from Lepper, Green, and Nisbett's research, feelings of being powerful and smart (intrinsic motivators) are more likely to keep people playing than extrinsic rewards.

34.5.4 Collecting and Completing

The chief goal of many games is to acquire all of the available items, rewards or knowledge contained in the game. In games such as *Pokemon Crystal* (Nintendo Japan 2000) the central challenge is to acquire as many of the Pokemon characters as you can and learn all of their skills well enough to outsmart your opponent at selecting the right characters for a head-to-head competition. Not coincidentally, the catch phrase for the *Pokemon Crystal* game is "Gotta catch 'em all!"

This game mechanic is also used by numerous games to add depth and repeat play. Though this isn't the primary mechanic in *Madden NFL 06* (Electronic Arts Inc. 2005), the ability to collect electronic player cards and use them strategically in games provides incentive for gamers to experiment with much of the content that they may not experience if playing through a standard season.

34.5.5 Story

Characters and narrative help gamers attach meaning and significance to action sequences. There are those in the games industry that propose that many games neither have nor need a story. It is our contention that the key to understanding narrative in games is to realize that storylines may be both embedded in the game, or they may emerge in the course of playing a game.

When most consumers think about story, they think about embedded storylines (Levine 2001). *Final Fantasy X* (Square Co., Ltd. 2001) tells its story by cutting action sequences with a series of full-motion cut scenes, real-time cut scenes and player-driven character interactions. The embedded story forms a central part of the appeal of the game. But Ken Levine (2001) points out that much of the narrative in a game emerges in the successes and failures experienced by players throughout the course of the game. This is especially true of multiplayer games, in which story is often generated exclusively by the interactions of the participants. As Levine describes it, the story is generated by replaying an abstract narrative structure with a strict set of rules and possibilities within a novel set of circumstances. Sporting events both within and outside of the video game world provide an excellent example of this type of narrative. No author scripted the result of the last World Cup tournament, but each such event has the potential to be an epic narrative for both participants and viewers.

34.5.6 Technological Innovation Drives Design

There is a great deal of pressure on designers to utilize new technologies that may break old interaction models. The desire to experience ever more realistic and imaginative games has pushed game developers into engineering and computer science research domains. Likewise, technology often drives game design in order to showcase new capabilities. The constant demand for novelty can be strong enough incentive for game makers to try untried designs, "spruce up" familiar interfaces and break rules of consistency. For example, the original *NFL2K* series (Sega 1999) sported a new interface model in which users selected interface areas by holding the thumbstick in a direction while pressing a button. It is possible that Sega chose the new design primarily because it was new and different from existing interfaces. It required the somewhat new (at that time) mechanics of the thumbstick on the controller, it minimized movement because one could point directly at any given item in the menu, and it was cleverly shaped like a football (which made more sense for *NFL2K* than for another sport). However, errors due to more error-prone targeting and the hold and click interaction metaphor made the system harder for first-time players to use.

34.5.7 Perceptual-Motor Skill Requirements

The way that functions are mapped onto available input devices can determine the success or failure of a game. A crucial part of the fun in many games comes from performing complex perceptual-motor tasks. While today's arcade-style games include increasingly more sophisticated strategic elements, a core element of many games is providing the ability to perform extremely dexterous, yet satisfying physical behaviors. These behaviors are usually quick and well-timed responses to changes and threats in the environment. If the controls are simple enough to master and the challenges increase at a reasonable difficulty, these mostly physical responses can be extremely satisfying (Csikszentmihalyi 1990).

Problems can arise when games require unfamiliar input devices. This is a common complication in console game usability research because new console systems usually introduce new input device designs unique to that system (see Figure 34.1). Furthermore, game designers often experiment with new methods for mapping the features of their games to the unique technical requirements and opportunities of new input devices. Unfortunately, this does not always result in a better gameplay experience.

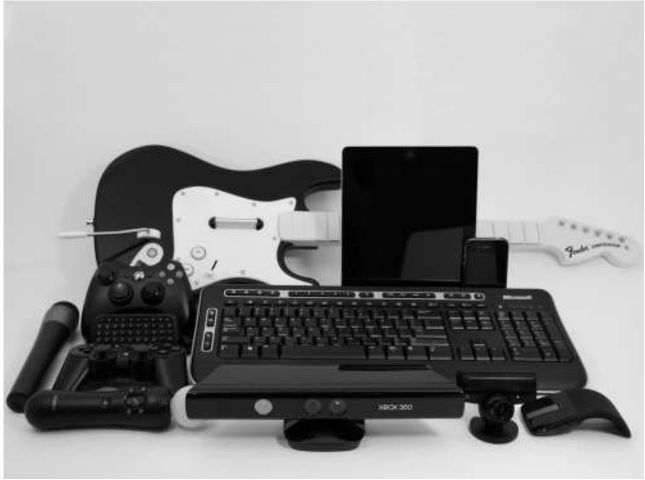


FIGURE 34.1 This is a sample of a variety of input devices used in games from the traditional keyboard and mouse to more advanced input mechanisms.

34.5.8 Balancing Multiplayer Games

As we've seen, different strategies can be used to support a broad range of player skill levels. While this is complicated in single-player games, it becomes even more daunting when players of different skills make up both the opponents and the allies. By far the most common strategy for regulating the challenge level of online multiplayer games is to provide strong matchmaking tools. *Internet Backgammon* (Microsoft Windows XP 2001) automatically matches players from around the world based on their self-reported skill level. Other games use algorithms that count player winning percentages and strength of opponent.

Another strategy seeks to solve skill balance problems outside of the game by offering players a wide array of arenas and/or game types. Rather than actively connecting players of like skill, this approach provides a broad array of places to play and allows players to self-select into game types that suit their style and attract the players with whom they want to associate. A final approach, used frequently by instant messaging clients (e.g., Yahoo IM, Windows Live Messenger), is to make it easy to start a game with friends. Though skill levels of friends may not always match, you are presumably less likely to perceive the match as unfair against people whom you know.

Game designers also employ a variety of in-game strategies to balance out the competition. The most common way to allow less-skilled players to compete effectively with more skilled players is to play team games. Many games allow players to take on a variety of roles. Capture the Flag is a common backyard tag-style game that forms the basis for a game type in many first-person shooters. Some players go on offense to capture the opponent's flag, some stay back to defend their flag. Likewise, some players may take a long-range sniper weapon to frustrate the enemy, while others take weapons that are more effective at close range. The defending and sniping roles can be more comfortable for some novice players because they allow the player to seek protective cover, they require less knowledge of the play field and one-on-one combat can be avoided.

Some first-person shooters also allow the game host to set "handicaps" for successful players and bonuses for less successful players. For example, one version of *Unreal Tournament* increased the size of the player's character with every point that they won, and decreased the size of the character every time that the character died. The size differences made successful players easier targets and unsuccessful players more difficult targets.

34.5.9 Enabling Social Networking

Skill level is not the only dimension of importance. Systems such as the Xbox Live service seek to group players of similar attitudes and behaviors. The goal is to place more antagonistic players with like-minded others who appreciate tough talk, while protecting players who don't want to or shouldn't be exposed to aggressive or offensive language.

This is just one of many tools that games have developed to promote group play and improve communication between online players. Most online games include some form of in-game messaging. Often this messaging is tailored for fast and efficient communication of key functions or timely game events. Most MMORPGs have negotiation systems that help people trade objects safely and efficiently. Most first-person shooters present automatic messages telling you which players have just been killed by whom. Real-time strategy (RTS) games allow gamers to set flares or markers on the map to notify your allies of key positions. Each game genre has a key set of in-game communication tools to support the game play.

More and more frequently game designers are starting to embed cooperative tools into the environment itself. Much of the benefit of vehicles in first-person shooter games comes from their use as cooperative tools. Used in concert these tools can often be extremely effective. From a game design perspective, they provide a great incentive for people to come together, strategize and work cooperatively. These shared successes can be a huge part of the fun in online games.

Massively multiplayer games employ a wide variety of incentives for players to form groups. Success comes from taking on missions that are far too dangerous for any single player to accomplish on their own. Players choose a role and learn to compliment the strengths of the other members of their group in order to overcome fearful enemies and accomplish great quests. Recent massively multiplayer games have invested even more in the creation of large-scale guilds. Most massively multiplayer games intentionally keep large areas of information secret from their players to provide the need and the opportunity for symbiotic relationships between experts and novices. In addition, modern MMO games often provide pyramid-scheme style bonuses in wealth and experience to the leaders and captains of guilds. In return, members receive physical, material, educational, and social protection from their leaders and peers.

34.5.10 Mobile Gaming Environmental Factors

The success of *Tetris* points out several fundamental truths of portable gaming. The most successful portable games fit efficiently and effectively into the user's lifestyle. They don't require prolonged concentration and are robust to environmental distractions, allowing the busy person to pause when needed and resume without difficulty. This is why so many successful portable games are brief, turn-based, or pauseable. They're with you when you have time to play—on the bus to school, or train to work, in the back of a car on the way to practice, or at a friend's house for head-to-head competition. They're challenging, but they don't require special experience or knowledge to be successful at the start of the game. Many portable successes are simple, familiar or particularly resistant to creating failure states.

Many of the same fundamental truths apply to mobile phone games. Mobile games appear poised to break new ground in social gaming due to increased accessibility to communication, connectivity, location, and identity features. The barriers to mobile phone gaming come largely from lack of hardware and software standardization. There are dozens of mobile game development platforms and thousands of hardware form factors. For example, while iPhone has a substantial amount of mind-share in the mobile space, it was estimated that there were a little over 6 million of them in operation in the United States in 2009 (Nielsenwire 2009).

34.6 IMPORTANT FACTORS IN GAME TESTING AND EVALUATION

Most game genres are subtly different in the experiences that they provoke. It may seem obvious that the point of game design is making a fun game. Some games are so fun that people will travel thousands of miles and spend enormous amounts of money to participate in gaming events. However, we would like to propose a potentially controversial assertion: the fundamental appeal of some games lies in their ability to challenge, to teach, to bring people together, or to simply experience unusual phenomena. Likewise, the definition of fun may be different for every person. When you play simulation games, your ultimate reward may be a combination of learning and mastery. When you play something like the *MTV Music Generator* (Codemasters 1999), your ultimate reward is the creation of something new. When you go online to play card games with your uncle in Texas, you get to feel connected. *Flight SimulatorX* (Microsoft Corporation 2006) lets people understand and simulate experiences that they always wished they could have. While these may be subcomponents of fun in many cases, there may be times when using "fun" as a synonym for overall quality will lead to underestimations of the quality of a game. While a fun game is often synonymous with a good game, researchers are warned to wisely consider which measures best suit the evaluation of each game that they evaluate.

34.6.1 Game Designer Intent

As stated earlier, the design goal of a game is to create a pleasurable experience and to help users have fun, and the main user goal is to have fun. Goals in a game are not necessarily derived from external user needs as in productivity applications. In games, goals are defined in accordance with the game designer's vision, which is a novel position because historically, success in productivity application testing has been defined by the accomplishment of user tasks and goals (Pagulayan, Gunn, and Romero 2006). When approaching a game for UCD or testing, it is best to assume the role of facilitating the designer's vision for the game (Pagulayan and Steury 2004; Pagulayan et al. 2003) because many times, it is only the designer who can recognize when the player experience is not being experienced as intended. In traditional usability testing, it is often very recognizable when there is user error, but not so in games.

Davis, Steury, and Pagulayan (2005) discuss a case study in the game *Brute Force* (2003), which revealed that players were not encountering certain gameplay features early enough in their gameplay experience. It was not the case that players were failing, but that players were taking much longer to play through the second mission than intended. By understanding the design vision, the authors were able to work with the designers to provide feedback which resulted in shortening the second mission, and also reordering other missions to match the design intent of the game.

34.6.2 Ease of Use

The ease of use of a game's controls and interface is closely related to fun ratings for that game. Think of this factor as a gatekeeper on the fun of the game. If the user must struggle or cannot adequately translate their intentions into in-game behaviors, they will become frustrated. This frustration can lead the user to perceive the game as being unfair or simply inaccessible (or simply not fun). Thus, it becomes very clear why usability becomes very important in games. Ease of use should be evaluated with both usability and attitude measurement methodologies, which are discussed later in the chapter.

34.6.2.1 Basic Mechanics

The basic mechanics of a game are best imagined by this example. In chess, each player expresses more strategic desires by turn-taking, movement, checking, and capturing. These actions are the core mechanics. Combined with the board and pieces, roles and rules, and strategies and situations, they make up the experience of chess. Of these mechanics, movement is one of the most important; yet, it is simple. Each piece has a role with defined movements, but it also contains the magic of chess. The power of the queen is expressed by the lack of restraint on her movement. Every game, similarly, has a set of core mechanics. Getting the core mechanics right is fundamental to making a great game.

34.6.2.2 Starting a Game

Starting the kind of game that the user wants is an easily definable task with visible criteria for success. This is something one can measure in typical usability laboratories. Though designers often take game shell (the interface used to start the game) design for granted, a difficult or confusing game shell can limit users' discovery of features and impede their progress toward enjoying the game. The most immediate concern for users can be starting the kind of game that they want to play. Games often provide several modes of play. When the game shell is difficult to navigate, users may become frustrated before they have even begun the game. For example, we have found that many users are unable to use one of the common methods that sports console games use to assign a game controller to a particular team. This has resulted in many users mistakenly starting a computer versus computer game. Depending on the feedback in the in-game interface, users may think that they are playing when, in fact, the computer is playing against itself! In these cases users may even press buttons, develop incorrect theories about how to play the game, and become increasingly confused and frustrated with the game controls. The most effective way to avoid these problems is to identify key user tasks and usability test them.

Pagulayan et al. (2003) discuss a case study where they found issues with difficulty settings in the game shell. In early usability testing, participants were having problems with setting the difficulty level of opponents in *Combat Flight Simulator* (Microsoft Corporation 1998). This is a case where the users' gameplay experience would have been quite frustrating because of a usability error in the game shell if it were not addressed.

34.6.2.3 Tutorials or Instructional Gameplay

Tutorials are sometimes necessary to introduce basic skills needed to play the game. In this situation, instructional goals are easily translated into the usability lab with comprehension tasks and error rates.

One of the risks of not testing tutorials or instructional missions is inappropriate pacing, which can often result from an ill-conceived learning curve at the start of the game. Many games simply start out at too difficult a challenge level. This is an easy and predictable trap for designers and development teams to fall into because when designers spend months (or even years) developing a game, they risk losing track of the skill level of the new player. A level that is challenging to the development team is likely to be daunting to the beginner.

Unfortunately, the reverse can also be troubling to the new user. Faced with the task of addressing new players, designers may resort to lengthy explanations. Frequently, developers will not budget time to build a gradually ramping learning process into their game. Instead, they may realize late in the development cycle that they need to provide instruction. If this is done too abruptly, the learning process can end up being mostly explanation, and to be frank, explanation is boring. The last thing that you want to do is to bore your user with a longwinded explanation of what they are supposed to do when they get into your game. It is best to learn in context and at a measured pace or users may just quit the game.

A very positive example is the first level of *Banjo Kazooie* (Nintendo of America 2000). At the start the player is forced to encounter a helpful tutor and listen to a few basic objectives. Then they must complete some very basic objectives that teach some of the basic character abilities. Much of the tutorial dialogue may be skipped, but the skills necessary to continue must be demonstrated. In this way, the game teaches new skills but never requires tedious instruction. The player learns primarily by doing. All of this is done in the shadow of a very visible path onto the rest of the game so the user never loses sight of where they need to go.

34.6.2.4 Camera

The camera perspective (i.e., the view that the player sees into the virtual world) in a game is often treated as an advanced game mechanic. That is, known difficulties in seeing the environment, threats, and opportunities can be exploited to create challenge and tension. When not done effectively, this can result in a poor experience for many users who are powerless to see something that they believe they could see in real life. This can be frustrating, resulting in a loss of immersion in the game world.

A 3D isometric view is a very effective camera perspective for viewing game boards or maps or moving a character through a game world. But there is an important trade-off to consider when thinking about how much distance to have between the camera and the objects of view. The farther away the camera is, the more environment the user can consider in his or her strategy. For this reason, it's generally easier to drive a car, command troops, or play *Monopoly* from a bird's eye view. On the other hand, the user may lose a very important part of the visceral experience by being too far away. Viewing a race from the inside of the car will cut down on the user's awareness of competitor cars and upcoming turns, but it feels a lot faster and more intense. Because of this inherent trade-off and different preferences around it, most racing games allow the user to choose between several viewing distances. The same effect can be seen in a shooting game. *Gears of War* feels extra gritty and dangerous in part due to the intentional placement of the camera lower over the shoulder of the main character than in many shooting games (Microsoft Game Studios 2006).

It's also worth noting that tight spaces can cause havoc for over-the-shoulder cameras. User research can identify problem areas and suggest custom cameras to help users see what they are supposed to see. Because designers know, and professional software testers learn, where to go and what to avoid, UCD can be a necessary way to discover camera blind spots that can be extremely frustrating to the mass of users who only play through an experience once.

At the extreme end, with very good tuning, nontraditional camera perspectives can be part of the fun of a game. Crash Bandicoot successfully kept the game fresh and increased dramatic tension by switching the camera after several levels to look directly into frightened Crash's eyes as he ran from a charging dragon (Universal Interactive Studios 2001). Likewise, *Resident Evil* increased suspense and fear in its games by employing fixed cameras at awkward locations and forcing people to enter areas and fight zombies blindly (Capcom 2005).

34.6.2.5 In-Game Interfaces

In-game interfaces are used primarily to deliver necessary status feedback and to perform less-frequent functions. We measure effectiveness with more traditional lab usability testing techniques and desirability with attitude measurements such as surveys (e.g., see Section 34.6.2.6).

Some PC games make extensive use of in-game interfaces to control the game. For example, simulation and RTS games can be controlled by keyboard and mouse presses on interface elements in the game. Usability improvements in these interfaces can broaden the audience for a game by making controls more intuitive and reducing tedious aspects of managing the game play. In-game tutorial feedback can make the difference between confusion and quick progression in learning the basic mechanisms for playing. In this situation, iterative usability evaluations become a key methodology for identifying problems and testing their effectiveness (e.g., see Section 34.6.2.6).

Many complex PC and console video games make frequent use of in-game feedback and heads-up displays (HUD) to display unit capabilities and status. For example, most flight combat games provide vital feedback about weapons systems and navigation via in-game displays. Without this feedback, it can be difficult to determine distance and progress toward objectives, unit health and attack success. This feedback is crucial for player learning and satisfaction with the game. With increasing game complexity and 3D movement capabilities, these displays have become a crucial part of many game genres. Usability testing is required to establish whether users can detect and correctly identify these feedback systems. See Pagulayan et al. (2003) for detailed example of usability testing the HUD in *MechWarrior 4: Vengeance* (Microsoft Corporation 2000).

34.6.2.6 Mapping Input Devices to Functions

A learnable mapping of buttons, keys or other input mechanisms to functions is crucial for enjoying games. We measure effectiveness with usability techniques and desirability with attitude measurements. Without learnable and intuitive controls, the user will make frequent mistakes translating their desires into onscreen actions. We have seen consistently that these kinds of mistakes are enormously frustrating to users, because learning to communicate one's desires through an eight-button input device is not very fun. The selection of keys, buttons, and other input mechanisms to activate particular features is often called "control-mapping." Players tend to feel that learning the control-mapping is the most basic part of learning the game. It is a stepping stone to getting to the fun tasks of avoiding obstacles, developing strategies, and blowing things up.

By contrast with other ease of use issues, evaluating the control-mapping may involve as much subjective measurement as behavioral observation. Button presses are fast, frequent, and hard to collect automatically in many circumstances. Furthermore, problems with control-mappings may not manifest themselves as visible impediments to progress, performance, or task time. Instead, they may directly influence perceptions of enjoyment, control, confidence, or comfort. Due to differences in experience levels and preferences between participants, there may also be significant variation in attitudes about how to map the controls.

Dissatisfaction with the controller design can also be a central factor that limits enjoyment of all games on a system. For example, the results of one whole set of studies on the games for a particular console game system were heavily influenced by complaints about the system's controller. Grasping the controller firmly was difficult because users' fingers were bunched up and wrists were angled uncomfortably during game play. Ratings of the overall quality of the games were heavily influenced by the controller rather than the quality of the game itself.

Because of these concerns and the importance of optimizing control-mappings, we recommend testing control-mappings with both usability and attitude assessment methodologies.

34.6.3 Challenge

Challenge is distinct from ease of use and is measured almost exclusively with attitude assessment methodologies. This can be a critical factor to the enjoyment of a game, and obviously can be highly individualized and is rightly considered subjective.

Consumers may have difficulties distinguishing the "appropriate" kinds of challenge that result from calculated level and obstacle design from the difficulty that is imposed by inscrutable interface elements or poor communication of objectives. In either case, the result is the same. If not designed properly, the player's experience will be poor. Thus, it is up to the user research professional to make measurement instruments that evaluate the appropriateness of the challenge level independently of usability concerns. In one example, Pagulayan et al. (2003) used attitude assessment methodologies to determine the final design of the career mode in *RalliSport Challenge* (Microsoft Corporation 2002). In this situation, finding a solution to the design problem was not necessarily related to ease of use issues, or other usability-related issues. The final design was based on what was most fun and appropriately challenging for users.

34.6.4 Pace

We define pace as the rate at which players experience new challenges and novel game details. We measure this with attitude measurement methodologies.

Most designers will recognize that appropriate pacing is required to maintain appropriate levels of challenge and tension throughout the game. You might think of this as the sequence of obstacles and rewards that are presented from the start of the game to the end. However, the way a designer will address pace will depend on a variety of issues, including game type, game genre, and their particular vision for the gameplay experience. In a tennis game, pace can be affected by a number of things, including the number of cut scenes in between each point, to the actual player and ball movement speed (Pagulayan and Steury 2004).

One group at Microsoft uses a critical juncture analogy to describe pacing. As a metaphor, they suggest that the designer must attend to keeping the user's attention at 10 seconds, 10 minutes, 10 hours, and 100 hours. The player can always put down the game and play another one, so one must think creatively about giving the user a great experience at these critical junctures. Some games excel at certain points but not others. For example, the massively multiplayer game may have the user's rapt attention at 10 seconds and 10 minutes. And the fact that hundreds of thousands pay $15 per month to continue playing indicates that these games are very rewarding at the 100-hour mark. But anyone who has played one of these games can tell you that they are extremely difficult and not too fun to play at the 10-hour mark. At this point, you are still getting "killed" repeatedly. That is no fun at all. Arcade games obviously take this approach very seriously. Though they may not scale to 100 hours, good arcade games attract you to drop a quarter and keep you playing for long enough to make you want to spend another quarter to continue.

Pacing may also be expressed as a set of interwoven objectives much like the subplots of a movie. Again, *Banjo Kazooie* provides an excellent example of good pacing. Each level in *Banjo Kazooie* contains the tools necessary to complete the major objectives. Finding the tools is an important part of the game. New abilities, objectives, skills, and insights are gradually introduced as the player matures. While progressing toward the ultimate goal (of vanquishing the evil witch and saving the protagonist's sister), the player learns to collect environmental objects that enable them to fly, shoot, become invincible, change shape, gain stamina, add extra lives, and unlock new levels. This interlocking set of objectives keeps the game interesting and rewarding. Even if one is unable to achieve a particular goal, there are always sets of sub goals to work on. Some of which may provide cues about how to achieve the major goal.

34.6.5 Summary

Attitude methodologies are better apt to measure factors such as overall fun, graphics, sound, challenge, and pace. The typical iterative usability is an exploratory exercise designed to uncover problem areas where the designer's intentions don't match the user's expectations, as a result, we typically choose to not use usability testing to assess "fun" or challenge. When attempting to assess attitudinal issues such as "overall fun" and "challenge" we make use of a survey technique that affords testing larger samples. Internally, we have adopted the term Playtest or sometimes Consumer Playtest for this technique. At the same time, we use typical iterative usability methods to determine design elements which contribute to or detract from the experience of fun.

34.7 USER RESEARCH IN GAMES

34.7.1 Introduction to Methods—Principles in Practice

In Sections 34.7.1 to 34.7.7, we propose various methodologies and techniques that attempt to accurately measure and improve game usability and enjoyment. Many of the examples are taken from techniques used and developed by Studios User Research at Microsoft Studios.

Our testing methods can be organized by the type of data being measured. At the most basic level, we categorize our data into two types: *behavioral* and *attitudinal*. *Behavioral* refers to observable data based on performance, or particular actions performed by a participant that one can measure. This is very similar to typical measures taken in usability tests (e.g., time it takes to complete a task, number of attempts it takes to successfully complete a task, and task completion). *Attitudinal* refers to data that represent participant opinions or views, such as subjective ratings from questionnaires or surveys. These are often used to quantify user experiences. Selection of a particular method will depend on what variables are being measured and what questions need to be answered.

Another distinction that is typically made is between *formative* and *summative* evaluations, which we apply to our testing methods as well. *Formative* refers to testing done on our own products in development. *Summative* evaluations are benchmark evaluations, either done on our own products, or on competitor products. It can be a useful tool for defining metrics or measurable attributes in planning usability tests (Nielsen 1993), or to evaluate strengths and weaknesses in competitor products for later comparison (Dumas and Redish 1999).

While these methods are useful, they do not allow us to address issues with extended gameplay, that is, issues that may arise after playing the game for a couple of days or more. This is problematic, because one of the key challenges in game design is longevity. With the competition, the shelf life of a game becomes very limited.

34.7.2 Usability Techniques

Traditional usability techniques can be used to address a portion of the variables identified as important for game design. In addition to measuring performance, we use many standard usability techniques to answer "how" and "why" process-oriented questions. For example, how do users perform an attack, or why are controls so difficult to learn? We use (1) structured usability tests, (2) rapid iterative testing and evaluation (RITE) (Medlock et al. 2002, 2005), and (3) other variations and techniques, including open-ended usability tasks, paper prototypes, and gameplay heuristics. For clarity of presentation, each technique will be discussed separately, followed by a case study. Each case study will only contain information pertinent to a specific technique, thus examples may be taken from a larger usability test.

34.7.2.1 Structured Usability Test

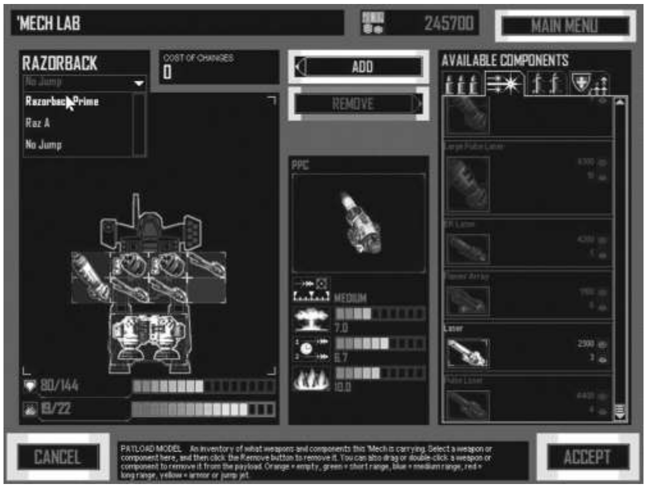
A structured usability test maintains all the characteristics that Dumas and Redish (1999) propose as common to all usability tests: (1) the goal is to improve usability of the product, (2) participants represent real users, (3) participants do real tasks, (4) participant behavior and verbal comments are observed and recorded, and (5) data are analyzed, problems are diagnosed, and changes are recommended. We have found that issues relating to expectancies, efficiency, and performance interaction are well suited for this type of testing. Some common areas of focus for structured usability testing are in game shell screens, or control schemes. The game shell can be defined as the interface in which a gamer can determine and or modify particular elements of the game. This may include main menus and options screens (i.e., audio, graphics, controllers, etc.).

An example which uses this method is in the *MechCommander 2* (*MC2*) (Microsoft Corporation 2001) usability test. Portions of the method and content have been omitted.

34.7.2.1.1 Case Study: MechCommander 2 Usability Test

MC2 is a PC RTS game where the gamer takes control of a unit of *mechs* (i.e., large giant mechanical robots). One area of focus for this test was on the *Mech Lab*, a game shell screen where mechs can be customized (see Figure 34.2). Here the gamer is able to modify weaponry, armor, and other similar features, and are limited by constraints such as heat, money, and available slots.

The first step in approaching this test was to define the higher order goals. Overall, the game shell screens had to be easy to navigate, understand, and manipulate, not only for those familiar with mechs and the mech universe, but also for RTS gamers who are not familiar with the mech universe.



PAYLOAD MODEL. An inventory of what weapons and components this 'Mech is carrying. Select a weapon or component here, and then click the Remove button to remove it. You can also drag or double-click a weapon or component to remove it from the payload. Orange = empty, green = short range, blue = medium range, red = long range, yellow = armor or jump jet.

FIGURE 34.2 Screenshot of the *Mech Lab* in *MechCommander 2*.

Our goal was for gamers to be able to modify/customize mechs in the *Mech Lab*.

As mentioned, one of the most important steps in this procedure is defining the participant profile(s). Getting the appropriate users for testing is vital to the success and validation of the data since games are subject to much scrutiny and criticism from its gamers. To reiterate, playing games is a choice. For *MC2*, we defined two participant profiles which represented all of the variables we wanted to cover. The characteristics of interest included those who were familiar with RTS games (experienced gamers) and those who were not RTS gamers (novice gamers). We also wanted gamers that were familiar with the mech genre, or the mech universe. Overall, we needed a landscape of gamers that had some connection or interest that would make them a potential consumer for this title, whether through RTS experience, or mech knowledge.

Tasks and task scenarios were created to simulate situations that a gamer may encounter when playing the game. Most importantly, tasks were created in order to address the predefined higher order goals. Participants were instructed to talk aloud, and performance metrics were recorded (i.e., task completion, time). The following are examples from the usability task list.

1. Give the **SHOOTIST** jumping capabilities.

   This task allowed us to analyze participant expectations. To succeed in this task, one had to select a "CHANGE WEAPONS" button from a different game shell screen which brought them into the *Mech Lab*. If the task was to change a weapon on a mech, the terminology would probably have been fine. Thus, this task had uncovered two main issues: (1) could they get to the *Mech Lab* where you modify the mech, and (2) were they able to discover the process of modifying the mech. It was accurately predicted that gamers would have difficulties with the button terminology. Thus, that button was changed to "MODIFY MECH."

   To change the components (i.e., add the jump jets), participants could either select the item, and drag it off the mech, or select the item, and press the "REMOVE" button (see Figure 34.2). One unexpected issue that arose was that participants unknowingly removed items because the distance required for removing an item was too small. The critical boundary that was implemented was too strict. In addition, participants had difficulties adding items by dragging and dropping because the distance required for adding an item was too large (i.e., the item would not stay on the mech unless it was placed exactly on top of the appropriate location). Appropriate recommendations were made, and implemented.

2. Replace the **MG Array** with the **Flamer**.

   One of the constraints presented for modifying a mech was heat limit. Each weapon had a particular heat rating. For example, if the heat limit for a mech is 35, and the current heat rating is 32, only weapons with a rating of 3 or fewer could be added. In this task, the "Flamer" had a heat rating much larger than the "MG Array," thus making impossible to accomplish this task without removing more items. The issues here were the usability of the heat indicator, heat icons, and the discoverability of heat limit concept. None of the participants figured this out. Recommendations included changing the functionality of the Heat Limit Meter, and to add better visual cues to weapons that exceed the heat limit. Both of these changes were implemented.

34.7.2.2 Rapid Iterative Testing and Evaluation Method

Medlock et al. (2002, 2005) have documented another common usability method used by Studios User Research at Microsoft Studios, which they refer to as the RITE method. In this method, fewer participants are used before implementing changes, but more cycles of iteration are performed. With RITE, it is possible to run almost two to three times the total sample size of a standard usability test. However, only one to three participants are used per iteration with changes to the prototype immediately implemented before the next iteration (or group of one to three participants).

The goal of the RITE method is to be able to address as many issues and fixes as possible in a short amount of time in hopes of improving the gamer's experience and satisfaction with the product. However, the utility of this method is entirely dependent on achieving a combination of factors (Medlock et al. 2002, 2005). The situation must include (1) a working prototype, (2) the identification of critical success behaviors, important, but not vital behaviors, and less important behaviors, (3) commitment from the development team to attend tests and immediately review results, (4) time and commitment from development team to implement changes before next round, and (5) the ability to schedule and/or run new participants as soon as the product has been iterated. Aside from these unique requirements, planning the usability test is very similar to more traditionally structured usability tests.

It is very helpful to categorize potential usability issues into four categories: (1) clear solution, quick implementation, (2) clear solution, slow implementation, (3) no clear solution, and (4) minor issues. Each category has implications for how to address each issue. In the first category, fixes should be implemented immediately, and should be ready for the next iteration of testing. In the second category, fixes should be started, in hopes that it can be tested by later rounds of testing. For the third and fourth category, more data should be collected.

The advantage of using the RITE method is that it allows for immediate evaluation and feedback of recommended fixes that were implemented. Changes are agreed upon, and made directly to the product. If done correctly, the RITE method affords more fixes in a shorter period of time. In addition, by running multiple iterations over time we are potentially able to watch the number of usability issues decrease. It provides a nice, easily understandable, and accessible measure. In general, the more iterations of testing, the better. However,

TABLE 34.2

Age of Empires **Example Concepts and Behaviors Categorized into Three Concepts Using the Rapid Iterative Testing and Evaluation Method**

Essential Concepts/Behaviors	Important Concepts/Behaviors	Concepts/Behaviors of Lesser Interest
• Movement	• Queuing up units	• Using hotkeys
• Multiselection of units	• Setting gathering points	• Using mini-map modes
• “Fog of war”	• Garrisoning units	• Using trading
• Scrolling main screen via mouse	• Upgrading units through technology	• Understanding sound effects

Source: Medlock, M. C., D. Wixon, M. Terrano, and R. Romero. 2002. *Using the RITE Method to Improve Products: A Definition and a Case Study*. Orlando, FL: Usability Professionals Association. With permission.

this method is not without its disadvantages. In this situation, we lose the ability to uncover unmet user needs or work practices, we are unable to develop a deep understanding of gamer behaviors, and we are unable to produce a thorough understanding of user behavior in the context of a given system (Medlock et al. 2002, 2005).

The following example demonstrates how the RITE method was used in designing the *Age of Empires II: The Age of Kings* (Microsoft Corporation 1999) tutorial. Again, portions of the method and content have been omitted. See Medlock et al. (2002) for more details.

34.7.2.2.1 Case Study: Age of Empires II: The Age of Kings Tutorial

Age of Empires II: The Age of Kings (*AoE2*) is an RTS game for the PC where the gamer takes control of a civilization spanning over a thousand years from the Dark Ages through the late medieval period. In this case study, a working prototype of the tutorial was available, and critical concepts and behaviors were defined. Also, the development team was committed to attending each of the sessions. And, they were committed to quickly implementing agreed upon changes. Finally, the resources for scheduling were available. The key element in this situation for success was the commitment from the development team to work in conjunction with us.

In the *AoE2* tutorial, there were four main sections: (1) marching and fighting (movement, actions, unit selection, the “fog of war”),* (2) feeding the army (resources, how to gather, where to find), (3) training the troops (use of mini-map, advancing through ages, build and repair buildings, relationship between housing and population, unit creation logic), and (4) research and technology (upgrading through technologies, queuing units, advancing through ages). Each of these sections dealt with particular skills necessary for playing the game. In essence, the tutorial had the full task list built in. In the previous case study, this was not the case.

At a more abstract level, the goals of the tutorial had to be collectively defined (with the development team). In more concrete terms, specific behaviors and concepts that a gamer should be able to perform after using the tutorial were identified, then categorized into the three levels of importance: (1) essential behaviors that users must be able to perform without exception, (2) behaviors that are important, but not vital to product success, and (3) behaviors that were of lesser interest. Table 34.2 lists some examples of concepts and behaviors from each of the three categories. This is an important step because it indirectly sets up a structure for decision rules to be used when deciding what issues should be addressed immediately, and what issues can wait.

The general procedure for each participant was similar to other usability tests we often perform.

If participants did not go to the tutorial on their own, they were instructed to do so by the specialist.

During the session, errors and failures were recorded. In this situation, an error was defined as anything that caused confusion. A failure was considered an obstacle that prevented participants from being able to continue. After each session, a discussion ensued among the specialist and the development team to determine what issues (if any) warranted an immediate change at that time.

In order to do this successfully, certain things had to be considered. For example, how can one gauge how serious an issue is? In typical usability tests, the proportion of participants experiencing the error is a way to estimate its severity. Since changes are made rapidly here, the criteria must change to the intuitively estimated likelihood that users will continue to experience the error. Another thing to consider is clarity of the issue, which was assessed by determining if there is a clear solution. We have often found that if issues do not have an obvious solution then the problem is not fully understood. And finally, what errors or failures were essential, important, or of lesser interest. Efforts of the development team should be focused on issues related to the essential category when possible.

At this point we broke down the issues into three groups. The first group included issues with a solution that could be quickly implemented. Every issue in this group was indeed quickly implemented before the next participant was run. The second group consisted of issues with a solution that could not be quickly implemented. The development team began working on these in hopes it could be implemented for later

* The *fog of war* refers to the black covering on a mini-map or radar that has not been explored yet by the gamer. The fog of war “lifts” once that area has been explored. Use of the fog of war is most commonly seen in RTS games.

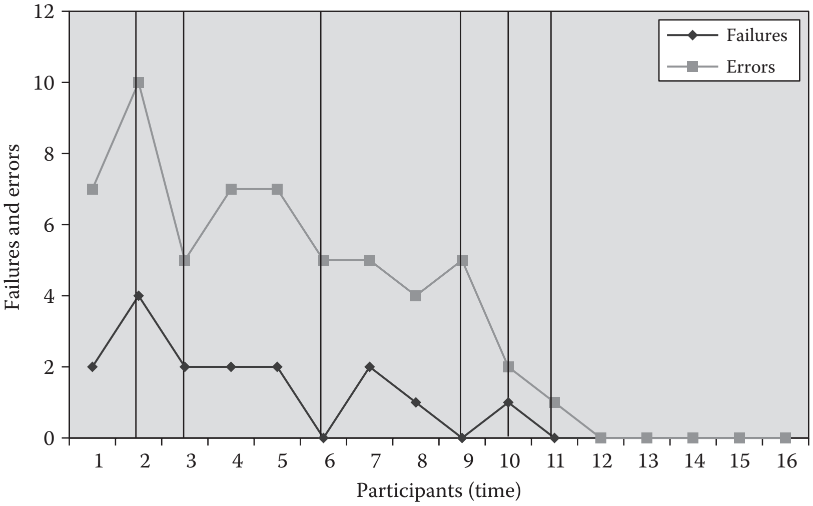


FIGURE 34.3 A record of failures and errors over time for the *Age of Empires* Tutorial when using the rapid iterative testing and evaluation method. (From Medlock, M. C., D. Wixon, M. Terrano, and R. Romero. 2002. *Using the RITE Method to Improve Products: A Definition and a Case Study*. Orlando, FL: Usability Professionals Association. With permission.)

iterations of the test. Finally, there were issues with no clear solutions. These issues were left untouched because more data were needed to assess the problem at a deeper level (i.e. more participants needed to be run). Any fixes implemented in the builds were kept as each participant was brought in. Thus, it was possible that many of the participants experienced a different version of the tutorial over the duration of testing.

Overall, seven different iterations were used across 16 participants. Figure 34.3 represents the number of errors and failures recorded over time. The number of errors and failures gradually decreased across participants as new iterations of the build were introduced. By the seventh and final iteration of the build, the errors and failures reliably were reduced to zero.

Although we feel that the *AoE2* tutorial was an enormous success, largely due to the utilization of the RITE method, the method does have its disadvantages and should be used with caution. Making changes when issues and/or solutions are unclear may result in not solving the problem at all, while creating newer usability problems in the interface. We experienced this phenomena a couple of times in the *AoE2* study.

Also, making too many changes at once may introduce too many sources of variability and create new problems for users. Deducing specifically the source of the new problem becomes very difficult. A related issue is not following up changes with enough participants to assess whether or not the solution really addressed the problem. Without this follow-up, there is little evidence supporting that the implementations made were appropriate (which is a problem with traditional usability methods as well). The last thing to consider is that other important usability issues that may surface less frequently are likely to be missed. Using such small samples between iterations allows for the possibility that those less occurring issues may not be detected.

34.7.2.3 Variations on Usability Methods

Now that we have presented two general types of usability testing, it is worth mentioning some variations on these methods: (1) open-ended tasks, (2) paper prototyping, and (3) gameplay heuristics.

In general, it is often recommended that tasks in usability tests be small, with a specified outcome (e.g., Nielsen 1993). However, we have found situations where the inclusion of an open-ended task yields important data as well. In many usability studies, we often include an open-ended task where participants are not instructed to perform or achieve anything in particular. In other words, there is no specified outcome to the participant. These tasks can be used to analyze how gamers prioritize certain tasks or goals in a nonlinear environment. These tasks are also useful in situations where structured tasks may confound the participant experience or situations where we are interested in elements of discovery. An example of an open-ended task is as follows:

> Play the game as if you were at home. The moderator will tell you when to stop.

This example was taken from a usability test on *Halo: Combat Evolved* (Microsoft Corporation 2001) (Pagulayan et al. 2003). Participants were presented with a mission with no instruction other than playing as if they were at home. Traditional usability metrics were not tracked or used. Instead, the focus was watching players and the tactics and strategies they employed while playing through the game. Results demonstrated that novice players would start firing at enemies as soon as they were visible, which was not how the designers intended combat to occur. The design intent was for combat to occur at closer ranges.

By allowing participants to play through the mission with no structured task, designers were able to detect the strategies players would employ. As a result, several changes were made to the gameplay to encourage players to engage in much closer combat. See Pagulayan et al. (2003) for more details.

Prototyping, heuristic evaluations, and empirical guideline documents are other techniques we often use when more time-consuming testing cannot be done. In practice, these techniques do not differ when used on games. Nielsen (1993) categorizes prototyping and heuristic evaluations as "discount usability engineering," and we would agree. We also tend to view empirical guideline documents in a similar manner. Empirical guideline documents are essentially lists of usability principles for particular content areas based on our collective experience doing user research. Examples of some of these content areas include console game shell design, PC game shell design, PC tutorial design principles, movement, aiming, and camera issues in first- or third-person shooter games, and online multiplayer interfaces. Desurvire, Caplan, and Toth (2004) have developed a list of heuristics targeted at computer and video games. These have been broken down into four general categories: game play, game story, game mechanics, and game usability. For the full list of gameplay heuristics, see Desurvire, Caplan, and Toth (2004).

34.7.3 Survey Techniques

The use of surveys has been explored in great depth (e.g., Bradburn and Sudman 1988; Couper 2000; Labaw 1981; Payne 1979; Root and Draper 1983; Sudman, Bradburn, and Schwarz 1996) and is considered a valid approach for creating an attitudinal data set as long as you ask questions that users are truly capable of answering (see Root and Draper [1983]). We conduct surveys in the lab and online in order to collect attitudinal data regarding gameplay experiences, self-reported play behaviors, preferences, styles, and motivations.

34.7.3.1 Formative Playtest Surveys

We combine hands-on play with surveys to create a formative lab method called "playtest." In nearly all instances, questioning follows the pattern of a forced choice question (sometimes a set of forced-choice questions) followed by a more open-ended question encouraging the participants to state the reasons behind their response. The goal of a playtest is to obtain specific information about how consumers perceive critical aspects of a game and provide actionable feedback to game designers (Davis, Steury, and Pagulayan, 2005; Amaya et al. 2008). The basic playtest method is used formatively to compare a product at time one and time two during its development. While a modified version of the basic method is used summatively to compare a summative evaluation to other, relevant games.

Playtests have several advantages and characteristics that differentiate them from usability studies and online survey research. For one, they focus mainly on perceptions. A larger sample size is required for statistical power and generalization of the results to the target audience. In most cases, 25–35 participants are used as a pragmatic trade-off between confidence and resource constraints. Specialized studies may require more. Statistical representations of the attitudinal measures are used to identify potential strengths and weaknesses that need to further be explored, provide information about the extent or severity of an attitude, or to describe trends of interest that come to light.

Conducting the studies in the lab, as opposed to online, allows us to test in-progress games during early development. It also allows us to take more control of the testing situations and monitor a limited amount of play behaviors that can be paired with the attitudinal data.

On the other hand, there are a few notable disadvantages. First, structured lab studies limit the amount of gameplay that can be tested in a single study. An artificial lab scenario or timeline may prevent the engineer from collecting critical information about the game experience. This is a classic validity trade-off present in most lab studies. Second, the amount of preparation that is required to design a playtest questionnaire takes some time. The study plan cannot be adjusted during the course of the study. The engineer must be intimately aware of the development team's questions to include all of the relevant questions in the survey. This requires domain knowledge in games and experience in survey design.

Second, the data can sometimes be difficult to interpret if the problem isn't major, or if the game facet is novel. Sometimes the data doesn't provide specific causes for the problem, making it difficult to nail down the best solution. Follow-up studies, (playtest or usability) are often recommended (or necessary) to parse out these issues. Finally, the approachable nature of the data is ripe for misinterpretation by team members who may not be familiar with the current state of the game, design details, or the methods used in playtest. It has been rightfully asserted that usability tests do not necessarily require a formal report upon completion (Nielsen 1993). We believe that may be less true for using this kind of technique. Careful and consistent presentation formats are required. Instructions and contextual information that will help with interpretation should be included, along with some form of qualitative data (e.g., open-ended comments). Metrics from games that are in development should focus on problem detection and often cannot be used for predictive purposes.

34.7.3.2 Development of the Playtest Surveys

Several steps have been taken to increase the effectiveness and efficiency of playtest. First, core questions for each genre, play style, and core game feature have been developed, using a multistep iteration and evaluation process. This process included construct development, survey item development, formative evaluation with subject matter experts, and formative evaluation of question validity using Cognitive Interviewing techniques and statistical validation. While many survey items have been validated, most playtests require new items to be created in order to meet the specific needs of that particular game and study goal. Therefore, a repository of previously used customized questions has been created, along with guidelines for writing new survey items.

34.7.4 Type of Formative Studies

There are essentially three types of formative studies that are conducted during development: (1) the critical facet test,

(2) the initial experience test, and (3) the extended playtest. We also run tests that do not easily fit into any of these categories, including subtle variations on the above, a few cases of large sample observationally based studies, and studies that focus on games from a more conceptual level.

As in Section 34.7.2, for clarity of presentation, each technique will be discussed separately, followed by a case study. Each case study will only contain information pertinent to a specific technique, thus examples may be taken from a larger playtest.

34.7.4.1 Critical Facet Playtest

Games often take the form of repeating a core experience within an array of different context and constraints. A driving game is always about aiming an object that is hurtling through space. The critical facet playtest focuses directly on that core experience and making sure that it is fun. While the core experience can often be assessed in usability testing, playtesting is necessary to assess attitudes and perceptions about the core experience.

The following example demonstrates how a critical facet test was applied to the critical facets of *Oddworld: Munch's Oddysee* (Microsoft Corporation 2001), an Xbox game. *Munch's Oddysee* is a platform/adventure game that allows you to switch back and forth between two main characters as they proceed through the increasingly difficult dangers of Oddworld on a quest to save Munch's species from extinction. The core gameplay is exploring the realm by running, jumping, and swimming through the environment. In this case, there were concerns about the user's visual perspective—which we typically call the camera. Does the camera support exploration of the environment?

34.7.4.1.1 Case Study: Munch's Oddysee, Camera

Previous usability testing with the *Munch's Oddysee* had determined that while some users indicated dissatisfaction with the behavior of the camera, other participants chose not to mention it at all while engaged in the open-ended usability tasks. The camera's behavior was programmed so as to create maximal cinematic effects (i.e., sometimes zooming out to show the size of an area) and also attempt to enhance gameplay. The camera would often show a specific view with the intent of showing you what was behind "the next door" or on the other side of the wall while still keeping the main character in view. While many users liked the look, style, and behavior of the camera, users often wanted more control over the behavior of the camera. Indeed some participants would actively say things such as, "That looks really cool right now [after the camera had done something visually interesting] but I want the camera back pointing this way now." Because feedback from the usability lab contained both positive and negative feedback, the development team didn't see the usability data as conclusive. Further, changing the camera would be a major cost to the game in terms of redesign and redevelopment time.

After having played the game for an hour, 25 participants were asked for general perceptions of the game. More specific questions followed. Questions related to the camera were asked in the latter portion of the questionnaire because previous experience in the usability lab had shown that merely mentioning the camera as part of a task would often cause participants previously silent on the subject to vociferously criticize aspects of the camera's behavior. With the knowledge that we wanted to factor out any priming-related effects, two analyses were conducted.

The first analysis was based on the response to the questions related to the behavior of the camera itself. Nearly half of the participants (46%) indicated that the camera did not give them "enough flexibility" of control.

The second analysis went back through individual responses to determine the attitudes of those participants who mentioned the camera before the survey first broached the subject. Forty-three percent of the participants were found to have mentioned the camera in a negative fashion prior to being asked specifically about the camera questions.

Based on this data and other anecdotal evidence, the development team chose to give the players more flexibility of camera control. The result was more frequent use of a camera behavior we termed a "3rd-person follow camera." The behavior of this camera had the double advantage of being more easily controlled by users and of conforming to a set of behaviors more often expected by users. It maintained focus on the main character without major adjustments to point of view (i.e., to "look over a wall" or "behind a door"). Other camera behaviors (i.e., still camera behaviors that pan with the character) are still a part of the game but have been localized to areas where these alternative camera behaviors can only create an advantage for the user.

34.7.4.2 Initial Experience Playtest

As with many things, first impressions are a key component of overall satisfaction with a game. Given that many games are a linear experience/narrative there is a lot of value in obtaining attitudinal data related to the first portions of gameplay. Lessons learned at first are often applied throughout the game. Obviously, the later portions of the game will never be experienced unless the first portions of the game are enjoyed.

The following example explains how a set of formative initial experience tests were run for *MC2*, the RTS game described earlier in the chapter. The earlier usability test focused on the choices that users could make prior to starting a mission, whereas this test focused on in-game components, such as the user's ability to take control of their squad and lead them through battles, while experiencing satisfaction and motivation to continue playing.

34.7.4.2.1 Case Study: MechCommander 2, Initial Missions

Twenty-five participants who were representative of the target market were brought onsite and were asked to begin playing from the first mission of the game. After each mission, participants were asked to stop playing and report their impressions of the "fun," the "excitement," and the "clarity" of objectives. Participants also indicated their "comfort level" with the basics of controlling the game. The participants were able to play through as many as three missions before the session ended. For purposes of brevity, this case study will focus on the results related to the first mission.

Although the first mission had been designed to offer tutorial elements in a brief and fun way, the experience was generally not satisfying to participants. Approximately a third of the participants had a poor initial impression of the game. They felt that the "challenge" was "too easy" and that the mission was "not exciting." Furthermore there were some problems related to clarity of goals. By combining the survey results with opportunistic observation, it was noted that some participants moved their units into an area where they were not intended to go. This disrupted their experience and limited their ability to proceed quickly. Responses to more open-ended questions indicated that some of the core gameplay components and unique features did not come across to users. Some users complained about the "standard" (predictable or commonplace) nature of the game. Finally several participants complained about the fact that they were "being taught" everything and wanted to "turn the tutorial off."

A number of actions were selected from team insight and user testing recommendations. First, the team decided to separate the tutorial missions from the required course of the game in order to save experienced players from the requirement of completing the tutorial. Second, the scope of the first mission was expanded so that users would have more time in their initial experience with the game. Third, the clarity of objectives was improved via minor interface changes. Fourth, addressing the same issue, the design of the "map" on which the mission took place was revamped to require a more linear approach to the first mission that limited the likelihood of users becoming "lost." Fifth, the amount and challenge of combat was increased. Finally, one of the unique components of the game was introduced. The new mission included the ability to "call in support" from off-map facilities. Despite all these changes, the mission was still targeted to be completed within approximately 10–15 minutes.

A follow-up playtest was intended to verify the efficacy of the changes. Twenty-five new participants were included in the study. Results from this test indicated that the design changes had created a number of payoffs. Far fewer participants felt the mission was "too easy" (13% vs. 33%), only 3% indicated that the mission was "not exciting," measures of "clarity of objectives" improved and, surprisingly, there was no drop-off on ratings of "comfort" with basic controls as a result of the tutorial aspects being improved. Results were not a total success, as some participants were now rating the mission as "too hard" while others in response to open-ended questions complained about the "overly linear" nature of the mission (i.e., there was only one way to proceed and few interesting decision related to how to proceed through the mission). In response to these results the development team decided to "re-tune" the first mission to make it a little easier, but not to address the comments related to the linearity of the mission. The second mission of the game was far less linear and so it was hoped that, with the major constraints to enjoyment removed, most participants would proceed to the second mission and find the mission goals to be more engaging. The data from mission 2 was rated far more "fun" than mission 1, validating their assumption.

34.7.4.3 Extended Playtests

While the initial experience with a game is very important, the success or failure of a game often depends on the entire experience. In an attempt to provide game developers with user-centered feedback beyond the first hour of game play, we conduct extended playtests that test several hours of consecutive gameplay. This is done by conducting playtest studies that run for more than 2 hours, or by having participants participate in more than one study. Attitudinal data is taken at significant time or experience intervals (i.e. after a mission). Participants can also provide self-directed feedback about a specific point in the game at anytime during the session. Basic behaviors and progress (e.g., completed level 4 at 12:51 PM) can be recorded by playtest moderators and used in conjunction with the attitudinal data.

34.7.4.4 Focus Groups

In addition to usability and survey techniques, we sometimes employ other UCD methods which are qualitative in nature. Focus groups provide an additional source of user-centered feedback for members of the project team. Our playtest facilities offer us the opportunity to precede a candid user discussion with hands-on play or game demonstration. In this setting, participants can elaborate on their experience with the product, provide feedback about the user testing process, speculate about the finished version of the product, generate imaginative feature or content ideas or find group consensus on issues in a way that models certain forms of real-world shared judgments.

For example, a focus group discussion regarding a sports game revealed that users had varying opinions about what it meant for a game to be a "simulation" versus an "arcade-style" game. Additionally, they had many different ideas as to what game features made up each style of game. From this conversation, we learned a few of the playtest survey questions were not reliable and needed revision.

Besides the general limitations inherent in all focus group studies (Greenbaum 1988; Krueger 1994), the focus groups described here typically include practical limitations that deviate from validated focus group methods. Generally, focus group studies comprise of a series of four to twelve focus groups (Greenbaum 1988; Krueger 1994), however, we typically run one to three focus groups per topic. This makes the focus group more of a generative research tool rather than a confident method for identifying "real" trends and topics in the target population. Fewer group interviews make the study more vulnerable to idiosyncrasies of individual groups.

34.7.5 Testing Physical Games

When testing physically-oriented games such as with Kinect, the Wii, or the Move, the basic concept of how to test is very similar to what we have noted previously. In a usability scenario we will use a mixture of open-ended and directed tasks and in a playtest scenario we will still ask users to play and then fill out brief surveys about the experience. The key considerations mostly focus on how the details of a test design will interact with the physical nature of the test. These differences touch on every aspect of research design.

34.7.5.1 Participants

When considering who to include in the study, some thought should be given to the preexisting skill level that you would prefer to see from your study participants. In the case of a singing game, the level of skill a participant brings may be a subject of interest prior to the test. And for a physically-oriented game, testing with people of different physical capabilities may be a matter of some importance.

You have to ensure that the participant is ready for and physically capable of participating in the study. Prior to the study, the participants might be asked to show up wearing clothes and shoes they can comfortably move around in. Liability issues may require you to modify any pretest agreements to cover potential physical issues that can occur.

A separate but related point is that the act of using a talent or physical capability, even in a private enclosed venue such as a research laboratory, may touch on feelings of anxiety and an individual's discomfort with their own ability to perform an action. While considerations like this are a factor in most research, the fact that a person is actually performing the action or skill-based task removes an abstraction layer that is more commonly a part of testing with standard game controller-based games. For example, if a player struggles with a game that requires expert use of a game controller then the player has a few strategies to manage self-esteem and express frustration. She has the option of choosing from statements such as "I cannot do that" or "the game is too hard" or "I am no good with this controller." Meanwhile in the context of a physical or talent-based game, while the same options are still available the "I cannot …" and "I am no good …" statements can point out deficits that are more personal in nature, are uncomfortable to discuss, and perhaps not pertinent to the needs of the test. Standard participant management techniques are called for to reduce anxiety and discomfort.

Finally, some consideration should be given to the idea of recruiting participants in groups. While not always true, it is very common for a physically oriented game to be oriented towards group play. The use of groups can complicate the assessment and that will be touched on in Section 34.7.5.3. But the use of groups also has the pleasant side effect of reducing some of the awkwardness of asking a user to perform a talent-based task and potentially failing in front of strangers.

34.7.5.2 Facilities

A key differentiator for physical-based gaming is that the facilities must support numerous scenarios. Sometimes the games are quite loud, in which case some soundproofing may be a helpful factor in the test setup. At other times the key factor would be providing a sufficient amount of space for two or more people to comfortably play together, while swinging their arms and perhaps moving through the rooms. Additional considerations should go to safety concerns; ensuring the floor surfaces in the testing environment are not and cannot become especially slick and to either move furniture out of the way or to ensure it is adequately padded in case of a fall.

From the perspective of the testing, the facilities should have multiple cameras watching the play experience, preferably from a frontal view, a side view and a top-down view. All views should have as wide an angle as possible available so that the tester can capture a reasonable range of motion without having to adjust the cameras in real time. Getting by without all the views is possible but the tester will miss details about the depth or actual size of a motion if only some of the fields of view are available.

34.7.5.3 Test Methods

For physical gaming, the primary adjustment to the methods occurs in usability testing. Put most simply, the think aloud technique may not be appropriate when the participant may be out of breath. It is more advisable to just concentrate on logging behaviors while a participant is playing and then to follow up with a more interview-oriented approach to understand what they were thinking at various points when they were playing.

Since the testing is of a physical experience, the researcher must divide attention between the game status as presented on screen to the participant and on the participants themselves as they move while playing. This only represents a change in that in physical scenarios there are numerous movements of interest, requiring multiple cameras to detect and study them.

From our experience, groups of participants are commonly treated as a 'single participant set'. So in the standard discussion of how many participants you want for a study (Nielsen and Landauer 1993) the number of individuals must be replaced with the number of groups. The plusses and minuses of using groups as the minimal measure of interest are best detailed within a context devoted to the subject. In our testing, to avoid problems associated with groupthink (Janis 1972) all participants are instructed to be open and honest and are informed that we are not trying to reach a consensus and that it is okay to disagree. These instructions are based on the standard approach to conducting a focus group (Greenbaum 1988; Krueger 1994). When coding the usability test, the behavior of the group and its success, failure and/or time on task is treated in similar fashion to coding the behavior of an individual. So if an 8-year-old child is on the cusp of figuring out an interface element but his mother insists that he is on the wrong track, then the nature of the failure is treated similarly to an individual going through the same process. The interface element is still judged to have failed the task, though the initial attractiveness is noted as well as the reason for group attention moving elsewhere.

34.7.5.4 Data Analysis and Reporting

The analysis of physically based gaming data is not especially different from the typical. Data can be coded and communicated similarly to data from other kinds of game testing. In the reporting, however there will often be a much stronger emphasis on visual analysis and accurately describing a physical motion. Time spent poring over video will likely increase and efficiencies can and must be sought from the use of strong video and data-coding software. Additionally the use of high-powered video editing software will often be required to create strong clear presentations of results.

34.7.6 Tracking Real-Time User Experience Instrumentation

For several years, we have been utilizing an instrumentation technique we call Tracking Real-Time User Experience (TRUE). TRUE instrumentation involves having the game track key user behaviors during gameplay, along with contextual variables, and allowing users to answer survey questions at key points in time. The TRUE method provides broad coverage, detailed recommendations, and quick turnaround of data collection, analysis, and recommendations. Its unique combination of quantitative analysis of behavioral data, collection and analysis of evaluative data (ratings and preferences), and ability to link all of these to qualitative data (video) make it a very powerful tool in a user researcher's arsenal. It combines the best of problem detection with in-depth analysis; the in-depth analysis makes the data actionable by design teams. The advantages of this combination deserve some discussion, and useful and practical recommendations for applying TRUE are discussed elsewhere (Kim et al. 2008; Schuh et al. 2008).

34.7.6.1 Behavior, Evaluation, and Context

To fully understand how effectively a game meets the intent of the designers one needs to know what people are doing at any point in the game (behavior), what conclusions they are drawing about the game (evaluations), and what in the mechanics of the game is leading to these behavioral and evaluative results (context). By collecting both behavior and evaluation and linking those results to indexed video, the development team can quickly identify (detect) problem areas and then zoom down to video to determine what elements of the game are creating those problems. Put another way, the TRUE method captures the primary elements needed to empirically understand and change the mechanics of a game in order to produce improved dynamics (behavior) and aesthetics (evaluation). Other methods (e.g., usability testing) may capture all these elements but they often do so in an informal way (thinking out loud and watching participants). In addition, the labor intensive nature of these data collection and analysis techniques means that, from a practical standpoint, they cannot cover the entirety of a sizable game. This increase in efficiency provided by TRUE does not just make work easier, it opens new possibilities. The scope of the application of TRUE combined with its breadth and depth make it unique.

34.7.6.2 Issue Discovery

Like other user research methods TRUE can be thought of as a "discovery" method. It collects and plots data so that anomalies and patterns can be spotted. Once an anomaly is spotted, for example, more deaths than expected at a given place or time or users not progressing past a given point in a mission or on a map, one can look at more and more detailed elements of the data until the likely cause for the unexpected outcome is discerned. This often involves a progressive drill down into the data, which may lead ultimately to viewing the video showing exactly what players did and in what context they did it. It shows what the team as a whole could not have predicted or at least did not anticipate. This "detection" quality is an essential characteristic of many user research methods (traditional usability test, RITE tests, Heuristic reviews, and others). TRUE is not really intended as a method to compare designs. Other methods like playtest and benchmark testing are better suited to those tasks.

Like other methods TRUE is best applied in some contexts and not at all applicable in some others. However the conditions for success for TRUE are more subtle than those of other methods. Specifically, TRUE is best applied when the development effort is characterized by common framework and shared definitions. For example, TRUE works well when missions are defined in a similar way throughout the game. Specifically, a mission begins when a prior climatic battle is won (e.g., the previous boss is vanquished) and ends when a final climatic battle is won (e.g., the current boss is killed). It also is well applied to games where the player navigates a map facing puzzles to solve and enemies to vanquish or in racing games where there is a clear start and finish of a gameplay segment. Its application is difficult if not impossible to a game where missions are defined in an inconsistent way, for example, sometimes a mission ends when a boss is vanquished and sometimes it ends when a boss is encountered. This kind of consistency needs to exist in all parts of the game. If the player relationship between damage and weapon use is not lawful and uniform, then analyzing the game with automated tools becomes difficult or impossible. The analyst is confronted with anomalies that defy explanation or are explained by lapses in consistency. Needless to say such haphazard elements will also make the game difficult to play and not in the good sense of that term. That is not difficult because the game is challenging but because it is arbitrary. That said, TRUE can detect bugs in a game—an accidently overly powerful enemy for example. Those kinds of bugs will stand out as anomalies just like a level or puzzle that is too challenging.

To be applied, the TRUE method must have a working prototype. Thus, it cannot be applied to an effort that is at the stage of ideation or story boards. It could be applied to any portion of a game that is playable. It could also be applied to a prior version of a game to identify areas for improvement. It can't be applied to other elements that contribute to the success of a game, such as story line or character depth. However, while these elements may play significant roles in a game's success, the game play is what defines a game and differentiates it from other art forms like books or movies.

In summary, TRUE is a powerful and flexible method, but requires a great amount of investment. When done correctly, its application has resulted in hundreds of improvements to dozens of games. It has greatly extended the "reach" of empirical methods into game design and development. It stands as a proven, practical, and effective method that has contributed to the development and commercial and critical success of many games (Thompson 2007). It represents a methodological breakthrough in games research and stands as a yardstick against which new methods can and should be assessed.

34.7.7 A Combined Approach

There are a variety of user research techniques we use in game development and evaluation—each suited for answering a particular type of question the development team has. As can be seen in this chapter, there are clear differences between the techniques but the end goal is the same: improve the user experience in our games. The important thing to realize is that no one method exists independent of other methods. The following is a brief case study on a PC game called *Rise of Nations: Rise of Legends* (Microsoft Game Studios 2006) that demonstrates how some of these techniques can be used in tandem.

34.7.7.1 Case Study: *Rise of Nations: Rise of Legends*

Rise of Nations: Rise of Legends is an RTS game in which the player must acquire resources, build up their armies and cities, and defeat/overtake any enemies in an effort to take control of the map. Our initial concern was the core controls—users must be able to successfully gather resources, build units and buildings, and engage in combat. A secondary concern was related to the Conquer the World Campaign and the mission progression throughout it.

The first issues addressed were the core controls and gameplay. We wanted to know if users could successfully build, resource, and engage in combat, so we decided to run a RITE usability test on the core gameplay and controls. This RITE testing was only possible because we had the development team from Big Huge Games on site with the ability to make quick iterative changes to the game between participants.

For the initial building controls, in order to place a building, the user had to determine where they wanted it, and then *right click* the mouse button to place the building on the map in that location. During our initial RITE study, we observed several participants attempting to place buildings using a *left click*. Moreover, they continued to make this error even after they discovered how to properly execute this action (right click). We attempted to fix this issue by adding a visual prompt when placing buildings (i.e., a visual icon of a mouse with the right button highlighted green) and tested this fix with four subsequent participants. All four understood the prompt; however, all continued to sporadically make left-click errors. We decided then to modify the controls to eliminate this usability issue and the game was changed to allow either left- *or* right-click building placement. No user following the change had difficulty placing buildings.

In addition to building, we were also interested in finding out if users could successfully resource. We noticed early on in the study that some participants did not explore the map at all. This type of behavior has repercussions for resourcing in that in order to find resources, one must explore the map. The designer intent for the game was that users would explore the map early on in order to find resources and plan strategies. It became clear that one of the barriers to exploration was that users needed to create scout units to help them explore. Yet, in order to create a scout unit, one had to first build a barracks. Both the time and sequence required for these events meant that users did not always create scout units and if they did, it was usually after a bit of time playing. The team decided to give users a “free scout unit” at the start of the game. The scout unit was placed conspicuously right in front of the player’s home city. As a result, fourteen of the next fifteen users explored the map and resourcing performance was greatly improved.

During this RITE study, we also discovered some small usability issues related to combat. These issues were addressed during the course of the study, changes were made to the game, and these changes were validated. Thus, by the end of the study, we were able to improve the usability of the core controls and gameplay and have those improvements implemented into the game.

After this initial RITE study, we began concentrating on the single-player Conquer the World Campaign. The Conquer the World Campaign involves the player progressing through 47 separate missions “scenarios” with the end goal of conquering the world map. The team wanted to apply the RITE techniques to each scenario in order to get fine-grained user data and determine if there were any problematic areas across the campaign. The design questions we wanted answered were the following: Where are users getting blocked? Where are users not having fun? Is the challenge ramp appropriate? Do users think a particular scenario is too easy or too difficult? What do users like/dislike about the campaign experience after several hours?

Some of these design questions we could not answer via usability techniques alone and RITE testing 47 missions was both time and resource prohibitive. Thus, we combined several research methodologies to help answer the design questions. We decided to get broad coverage of the game, follow it up with a fine-grained analysis of the problem areas, fix the issues, go back and cast the broad net, and follow that up by more focused fine-grained testing. To begin, we started running extended playtests over the weekends with TRUE instrumentation. The TRUE instrumentation gave us a mapping of user behavior over time as well as attitudinal data to couple with it. The Monday following the weekend test, we utilized the TRUE data to determine problematic areas to focus our RITE testing on during the week. During the week, we would RITE test those scenarios, all along fixing the builds, and the following weekend, run extended playtests with TRUE instrumentation on the updated game. Over the course of several weeks, we were able to get complete coverage of the entire Conquer the World Campaign and answer most of the design questions we set out to answer. By combining RITE testing, which allowed us to discover and fix many instances of user behavior not matching designer intent, and TRUE instrumentation, which allowed us to discover patterns in user behavior over the course of extended gameplay as well as giving us rich qualitative information and attitudinal data, we were able to help better match the user experience to the design vision across the entire game.

34.8 CONCLUSION

The need for the continued development of UCD methods in video games has indeed arrived. Games drive new technologies, generate enormous revenue, and affect millions of people. In addition, games represent a rich space for research areas involving technology, communication, attention, perceptual-motor skills, social behaviors, and virtual environments, just to name a few. It is our position that video games will eventually develop an intellectual and critical discipline, like films, which would result in continually evolving theories and methodologies of game design. The result will be an increasing influence on interface design and evaluation. This relationship between theories of game design and traditional human–computer interaction evaluation methods has yet to be defined, but definitely yields an exciting future.

UCD methods are beginning to find their way into the video games industry. Commercial game companies such as Ubisoft Entertainment and Electronic Arts, Inc., in addition to Microsoft, employ some level of user-centered methodologies to their game development process. Games share as many similarities as differences to other computer fields that have already benefited from current UCD methods. Thus, it makes sense to utilize these methods when applicable but also to adapt methods to the unique requirements that we have identified in video games.

In this chapter, we emphasized the difference between games and productivity applications in order to illustrate the similarities and differences between these two types of software applications. We also chose to reference many different video games in hopes that these examples would resonate with a number of different readers. Case studies were included to demonstrate, in practice, how we tackle some of the issues and challenges mentioned earlier in the chapter. It is our intention that practitioners in industry, as well as researchers in academia, should be able to take portions of this chapter and adapt them to their particular needs when appropriate, similar to what we have done in creating the actual methods mentioned in this chapter. That said, we are upfront that most, if not all, of our user research methods are not completely novel. Our user research methods have been structured and refined based on a combination of our applied industry experience, backgrounds in experimental research, and of course, a passion for video games. This allows us to elicit and utilize the types of information needed for one simple goal: to make the best video games that we can, for as many people as possible.

ACKNOWLEDGMENTS

We would like to thank Studios User Research at Microsoft Studios. In addition, we would like to express our gratitude to Paolo Malabuyo, Michael Medlock, and Bill Fulton for their insights and input on the creation of this chapter, and Kathleen Farrell and Rally Pagulayan for their editing assistance and reviewing early drafts. The views and opinions contained in this chapter are those of the authors and do not necessarily represent any official views of Microsoft Corporation.

REFERENCES

Age of Empires II: Age of Kings. 1999. *Computer Software*. Redmond, WA: Microsoft Coporation.

Alpine Racer. 1995. *Computer Software*. Tokyo, Japan: Namco.

Amaya, G., J. Davis, D. Gunn, C. Harrison, R. Pagulayan, B. Phillips, and D. Wixon. 2008. Games user research (GUR): Our experience with and evolution of four methods. In *Game Usability: Advice from the Experts for Advancing the Player Experience*, ed. K. Isbister, N. Schaffer, 35–64. Burlington, MA: Morgan Kaufmann Publishers.

Animal Crossing. 2002. *Computer Software*. Redmond, WA: Nintendo of America.

Banjo Kazooie. 2000. *Computer Software*. Redmond, WA: Nintendo of America.

Bradburn, N. M., and S. Sudman. 1988. *Polls and Surveys: Understanding What they Tell Us.* San Francisco, CA: Jossey-Bass.

Brute Force. 2003. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Business Week. 2006. *A Brief History of Game Console Warfare.* http://images.businessweek.com/ss/06/10/game_consoles/source/7.htm (accessed March 22, 2008).

Cassell, J., and H. Jenkins. 2000. Chess for girls? Feminism and computer games. In *From Barbie to Mortal Kombat: Gender and Computer Games*, ed. J. Cassell, and H. Jenkins, 2–45. Cambridge, MA: The MIT Press.

Chalk, A. 2007. Halo 3 Sets New First-Day Sales Record. http://www.escapistmagazine.com/news/view/77341-Halo-3-Sets-New-First-Day-Sales-Record. (accessed June 23, 2010).

Combat Flight Simulator. 1998. *Computer Software*. Redmond, WA: Microsoft Corporation.

Couper, M. P. 2000. Web surveys: A review of issues and approaches. *Public Opin Q* 64:464–94.

Crackdown 2. 2010. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Crash Bandicoot: The Wrath of Cortex. 2001. *Computer Software*. Los Angeles, CA: Universal Interactive Studios.

Crawford, C. 1982. *The Art of Computer Game Design.* Berkeley, CA: Osborne/McGraw-Hill.

Csikszentmihalyi, M. 1990. *Flow-The Psychology of Optimal Experience.* New York: Harper & Rowe.

Dance Dance Revolution. 1998. *Computer Software*. Tokyo, Japan: Konami.

Davis, J. P., K. Steury, and R. J. Pagulayan. 2005. A survey method for assessing perceptions of a game: The consumer playtest in game design. *Game Stud Int J Comput Game Res* 5(1). http://www.gamestudies.org/0501/davis_steury_pagulayan/ (accessed November 22, 2005).

Desurvire, H., M. Caplan, and J. Toth. 2004. Using heuristics to improve the playability of games. In *CHI 2004: Conference on Human Factors in Computing Systems.* Vienna, Austria: ACM's Special Interest Group on Computer-Human Interaction.

Diddy Kong Racing. 1997. *Computer Software*. Redmond, WA: Nintendo of America.

Domjan, M. 2010. *The Principles of Learning and Behavior.* Belmont, CA: Wadsworth.

Dumas, J. S., and J. C. Redish. 1999. *A Practical Guide to Usability Testing*, Rev. ed., Portland, OR: Intellect Books.

Entertainment Software Association (ESA). 2009. *2009 Sales, Demographics and Usage Data: Essential Facts About the Computer and Video Game Industry.* Washington, DC: Entertainment Software Association.

Facebook. 2010. *Facebook's Farmville Application Page.* Facebook:http://www.facebook.com/apps/application.php?id=102452128776 (accessed April 25, 2010).

Final Fantasy X. 2001. *Computer Software*. Tokyo, Japan: Square Co., Ltd.

Flight Simulator X. 2006. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Forza Motorsport 4. 2011. *Computer Software*. Redmond, WA: Microsoft Studios.

Game Boy. 1989. *Computer Hardware*. Redmond, WA: Nintendo of America.

Game Boy Color. 1998. *Computer Hardware*. Redmond, WA: Nintendo of America.

Gameboy Advance. 2001. *Computer Hardware*. Redmond, WA: Nintendo of America.

Gardner, J. 2009. *Futurology: FarmVille on Facebook.* London Today: http://www.thisislondon.co.uk/lifestyle/article-23749479-futurology-farmville-on-facebook.do (accessed October 11, 2009).

Gears of War. 2006. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Greenbaum, T. L. 1988. *The Practical Handbook and Guide to Focus Group Research.* Lexington, MA: D.C. Heath and Company.

Guitar Hero. 2005. *Computer Software*. Santa Monica, CA: Activision.

Halo 2. 2004. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Halo: Combat Evolved. 2001. *Computer Software*. Redmond, WA: Microsoft Corporation.

Harvest Moon. 2003. *Computer Software*. Redmond, WA: Nintendo of America.

Hyman, P. 2005. State of the industry: Mobile games. *Game Developer Mag* 12(14):11–6.

IDG Entertainment. 2005. *IDG Entertainment Casual Games Market Report.* Oakland, CA: IDG Entertainment.

International Game Developers Association. 2005. IDGA 2005 mobile games whitepaper. Presented at the Game Developers Conference 2005 by the Mobile Games SIG, San Francisco.

Internet Backgammon, Windows XP. 2001. *Computer Software*. Redmond, WA: Microsoft Corporation.

Jablonsky, S., and D. DeVries. 1972. Operant conditioning principles extrapolated to the theory of management. *Organ Behav Hum Perform* 7:340–58.

Janis, I. L. 1972. *Victims of Groupthink.* Boston, MA: Houghton Mifflin Company.

Jordan, P. W. 2000. *Designing Pleasurable Products: An Introduction to the New Human Factors.* Philadelphia, PA: Taylor & Francis.

Kent, S. L. 2000. *The First Quarter: A 25-year History of Video Games.* Bothell, WA: BWD Press.

Kim, J., D. Gunn, E. Schuh, B. Phillips, R. Pagulayan, and D. Wixon. 2008. Tracking real-time user experience (TRUE): A comprehensive instrumentation solution for complex systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* Florence, Italy: ACM.

Krueger, R. A. 1994. *Focus Groups: A Practical Guide for Applied Research.* Thousand Oaks, CA: Sage.

Labaw, P. 1981. *Advanced Questionnaire Design.* Cambridge, MA: Abt Books Inc.

Lepper, M. R., and T. W. Malone. 1987. Intrinsic motivation and instructional effectiveness in computer-based education. In *Aptitude, Learning and Instruction III: Conative and Affective Process Analyses*, ed. R. E. Snow, and M. J. Farr, 223–53. Hillsdale, NJ: Erlbaum.

Lepper, M., D. Greene, and R. Nisbett. 1973. Undermining children's intrinsic interest with extrinsic rewards. *J Pers Soc Psychol* 28:129–37.

Levine, K. 2001. New opportunities for storytelling. Paper presented at the Electronic Entertainment Exposition. Los Angeles, CA.

Madden NFL 06. 2005. *Computer Software*. Redwood City, CA: Electronic Arts Inc.

Malone, T. W. 1981. Towards a theory of intrinsic motivation. *Cogn Sci* 4:333–69.

Maximejohnson. 2010. Tetris atteint les 100 millions de téléchargements payants (et une petite histoire du jeu). maximejohnson: http://www.maximejohnson.com/techno/2010/01/tetris-atteint-les-100-millions-de-telechargements-payants-et-une-petite-histoire-du-jeu/ (accessed May 24, 2010)

Mazur, J. 2006. *Learning and Behavior*, 6th ed., Upper Saddle River, NJ: Prentice Hall.

MechCommander 2. 2001. *Computer Software*. Redmond, WA: Microsoft Corporation.

MechWarrior 4: Vengeance. 2000. *Computer Software*. Redmond, WA: Microsoft Corporation.

Medlock, M. C., D. Wixon, M. McGee, and D. Welsh. 2005. The rapid iterative test and evaluation method: Better products in less time. In *Cost Justifying Usability*, ed. G. Bias, and D. Mayhew, 489–517. San Francisco, CA: Morgan Kaufmann.

Medlock, M. C., D. Wixon, M. Terrano, and R. Romero. 2002. *Using the RITE Method to Improve Products: A Definition and a Case Study*. Orlando, FL: Usability Professionals Association.

Microsoft. 2007. Global Entertainment Phenomenon "Halo 3" Records More Than $300 Million in First-Week Sales Worldwide. Microsoft Newscenter: http://www.microsoft.com/presspass/press/2007/oct07/10-04Halo3FirstWeekPR.mspx (accessed June 15, 2010).

Moore, K., and J. Rutter. 2004. Understanding consumers' understanding of mobile entertainment. In *Proceedings of Mobile Entertainment: User-centred Perspectives,* ed. Moore, and Rutter, 49–65. University of Manchester: CRIC.

MPAA. 2009. *Theatrical Market Statistics.* Washington, DC: MPAA.

MTV Music Generator. 1999. *Computer Software*. Warwickshire, UK: The Codemasters Software Company Limited.

NFL2K. 1999. *Computer Software*. San Francisco, CA: Sega of America, Inc.

Nielsen, J. 1993. *Usability Engineering*. San Francisco, CA: Morgan Kaufmann.

Nielsen, J., and T. K. Landauer. 1993. A mathematical model of the finding of usability problems. In *Proceedings of ACM INTERCHI'93 Conference*, 206–13. Amsterdam: ACM.

Nielsenwire. 2009. iPhone Users Watch More Video… and are Older than You Think: http://blog.nielsen.com/nielsenwire/online_mobile/iphone-users-watch-more-video-and-are-older-than-you-think/ (accessed June 15, 2009)

Nintendo. 2005. 2005 Annual Report. http://www.nintendo.com/corp/report/NintendoAnnualReport2005.pdf (accessed March 22, 2008).

Nintendo DS. 2004. *Computer Hardware*. Redmond, WA: Nintendo of America.

Nintendo Wii. 2006. *Computer Hardware*. Redmond, WA: Nintendo of America.

Norman, D. 2005. *Emotional Design.* Cambridge, MA: Basic Books.

Oddworld: Munch's Odyssey. 2001. *Computer Software*. Redmond, WA: Microsoft Corporation.

Pagulayan, R. J., and K. Steury. 2004. Beyond usability in games. *Interactions* 11(5):70–1.

Pagulayan, R. J., D. V. Gunn, and R. L. Romero. 2006. A gameplay-centered design framework for human factors in games. In *2nd Edition of International Encyclopedia of Ergonomics and Human Factors*, ed. W. Karwowski, 1314–9. London, England: Taylor & Francis.

Pagulayan, R. J., K. Steury, B. Fulton, and R. L. Romero. 2003. Designing for fun: User-Testing case studies. In *Funology: From Usability to Enjoyment*, ed. M. Blythe, A. Monk, K. Overbeeke, and P. Wright, 137–50. Netherlands: Kluwer Academic.

Payne, S. L. 1979. *The Art of Asking Questions.* Princeton, NJ: Princeton University.

Playstation Move. 2010. *Computer Hardware*. Tokyo, Japan: Sony Computer Entertainment.

Pokemon Crystal. 2000. *Computer Software*. Tokyo, Japan: Nintendo Japan.

Preece, J., Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey. 1996. *Human-Computer Interaction*. Reading, MA: Addison-Wesley.

Project Natal. 2010. *Computer Hardware*. Redmond, WA: Microsoft.

RalliSport Challenge. 2003. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Ratchet and Clank Future: A Crack in Time. 2009. *Computer Software*. Tokyo, Japan: Sony Computer Entertainment.

Resident Evil 4. 2005. *Computer Software*. Osaka, Japan: Capcom.

Rise of Nations: Rise of Legends. 2006. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Rock Band. 2007. *Computer Software*. Redwood City, CA: Electronic Arts.

Root, R. W., and S. Draper. 1983. Questionnaires as a software evaluation tool. Paper presented at the ACM CHI, Boston, MA.

Saints Row 2. 2008. *Computer Software*. Agoura Hills, CA: THQ.

Salas, E., and D. Maurino. 2010. *Human Factors in Aviation,* 2nd ed., Burlington, MA: Elsevier.

Schuh, E., D. Gunn, B. Phillips, R. Pagulayan, J. Kim, and D. Wixon. 2008. TRUE instrumentation: Tracking real-time user experience in games. In *Game Usability: Advice from the Experts for Advancing the Player Experience*, ed. K. Isbister, and N. Schaffer, 237–65. Burlington, MA: Morgan Kaufmann Publishers.

Second Life. 2003. *Computer Software*. San Francisco, CA: Linden Research, Inc.

Shippy, D., and M. Phipps. 2009. *The Race for a New Game Machine.* New York: Kensington Publishing Corp.

Sidel, P. H., and G. E. Mayhew. 2003. The Emergence of Context: A Survey of MobileNet User Behaviour. www.mocobe.com/pdf/EmergenceofContext1.pdf (accessed June 15, 2010)

Singstar. 2004. *Computer Software*. Tokyo, Japan: Sony Computer Entertainment.

Sudman, S., N. M. Bradburn, and N. Schwarz. 1996. *Thinking about Answers: The Application of Cognitive Processes to Survey Methodology.* San Francisco, CA: Jossey-Bass.

Super Mario Bros Wii. 2009. *Computer Software*. Redmond, WA: Nintendo of America.

The Legend of Zelda: The Wind Waker. 2003. *Computer Software*. Redmond, WA: Nintendo of America.

Thompson, C. 2007. Halo 3: How Microsoft Labs Invented a New Science of Play. *Wired* 15:140–7.

Tom Clancy's Splinter Cell: Conviction. 2010. *Computer Software*. Paris, France: Ubisoft.

Tomb Raider: Underworld. 2008. *Computer Software*. London, England: Eidos.

Toy Soldiers. 2010. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Toy Soldiers: Match Defense. 2010. *Computer Software*. Redmond, WA: Microsoft Game Studios.

Wickens, C. D., and J. G. Hollands. 2000. *Engineering Psychology and Human Performance*, 3rd ed., Upper Saddle River, NJ: Prentice Hall.

Wii Sports Resort. 2008. *Computer Software*. Redmond, WA: Nintendo of America.

Yuen, M. 2005. The tipping point: The convergence of wireless and console/PC game design. *Game Developer Magazine*, September 1.

Part V

Designing for Diversity

35 Older Adults and Information Technology

Opportunities and Challenges

Sara J. Czaja and Chin Chin Lee

CONTENTS

35.1 INTRODUCTION

The expanding power and proliferation of information technologies such as the Internet and mobile devices have made possible for large numbers of people to have direct access to an increasingly wide array of information sources and services. Network usage is exploding, and new interfaces, search engines, and features are becoming available at an unprecedented rate. In 2010, about 79% of adults in the United States reported that they were Internet users and 66% have broadband access at home (Figures 35.1 and 35.2). In addition, 82% of Americans own a cell phone or some other device such as a Blackberry that is also used as a cell phone (Smith, 2010).

Use of technology is pervasive among all aspects of life and has become an integral component of work, education, communication, and entertainment. Technology is also being increasingly used within the health care arena for service delivery, in-home monitoring, interactive communication (e.g., between patient and physician), transfer of health information, and peer support. Use of automatic teller machines, interactive telephone-based menu systems, and digital entertainment equipment is also quite common. Furthermore, communication devices are becoming more integrated with computer network resources providing faster and more powerful interactive services. In essence, to function independently and successfully engage in routine activities, people of all ages will increasingly need to interact with some form of technology.

Coupled with the "technology explosion" is the aging of the population. In 2004, persons 65 years or older represented 12.4% of the U.S. population, and it is estimated that people in this age group will represent 20.6% of the population by 2050 (Figure 35.3). In addition, the older population itself is getting older. Currently, there are about 44.5 million people over the age of 75 years, and by the year 2050, there will be almost 50 million people 75+ years old and about 19 million people aged 85 and older (National Center for Health Statistics 2005; Federal Interagency Forum on Aging-Related Statistics 2010) (Figure 35.3).

Given that older people represent an increasing large proportion of the population and will need to be active users of technology, issues surrounding aging and information technologies are of critical importance within the domain of human–computer interaction (HCI). To ensure that older adults are able to adapt to the new information environment, we need to understand the following: (1) the implications of age-related changes in functional abilities for the design and implementation of technology systems; (2) the needs and preferences of older people with respect to the design

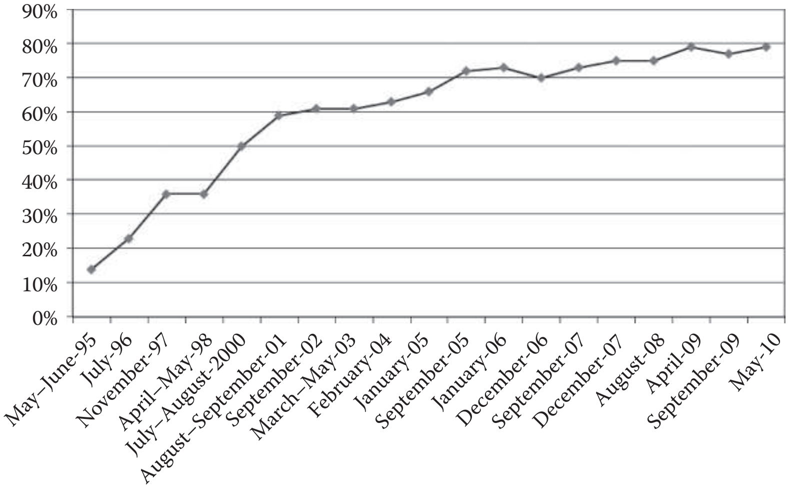


FIGURE 35.1 Percentage of adults online. (From Pew Research Center's Internet & American Life Project. http://www.pewinternet.org/Static-Pages/Trend-Data/Internet-Adoption.aspx. With permission.)

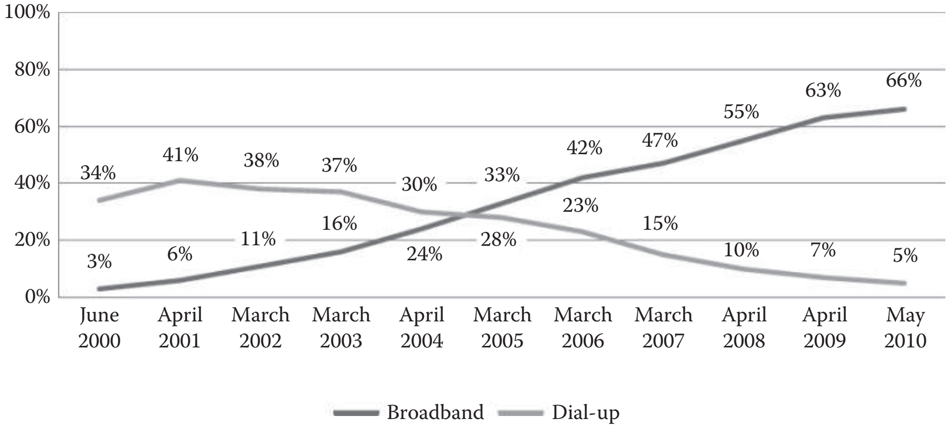


FIGURE 35.2 Percentage of adults who access Internet via broadband or dial-up. (From Smith, A. *Home Broadband 2010*. Pew Internet & American Life Project Surveys, 2010. http://pewinternet.org/Reports/2010/Home-Broadband-2010.aspx. With permission.)

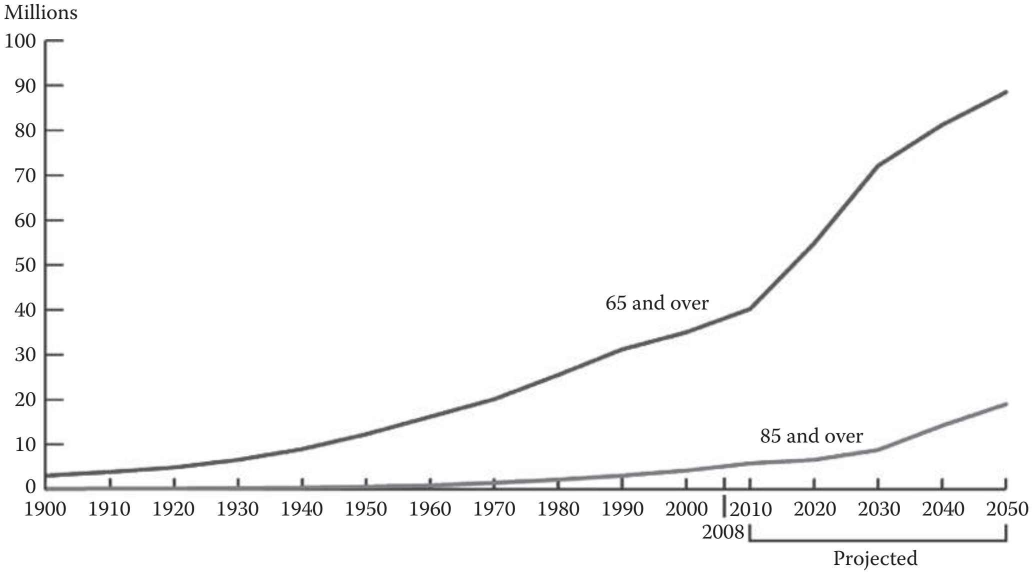


FIGURE 35.3 Percentage of U.S. population aged 65 and over. (From Federal Interagency Forum on Aging-Related Statistics. *Older Americans 2010: Key Indicators of Well-being*. Washington, DC: U.S. Government Printing Office, 2010. http://www.agingstats.gov/agingstatsdotnet/Main_Site/Data/2010_Documents/Slides/OA_2010.ppt.)

of technology interfaces and applications; (3) applications of technology that are useful to and usable by older adults; (4) the reasons for technology nonadoption among seniors; and (5) the problems and challenges that older adults confront when adopting new technologies.

The intent of this chapter is to summarize the current state of knowledge regarding information technologies and older adults. A particular focus will be on web-based technology applications. Topics that will be discussed include the following: adoption and use of information technologies by older adults, training, interface design issues, and other topics such as acceptance of by older adults, privacy, and trust in technology. A detailed discussion of the aging process will not be provided. There are many excellent sources of this material (e.g., Fisk et al. [2009]; Schulz et al. [2006]; Birren and Schaie [2001]). Instead, a brief review of age-related changes in abilities that have relevance to the design of technology systems will be presented. In addition, for the most part, material presented in the previous edition of this book (Czaja and Lee 2007) will not be reiterated. It is hoped that this chapter will serve to motivate researchers and system designers to continue to consider older adults as an important population within the HCI community.

35.2 USE OF TECHNOLOGY BY OLDER ADULTS

35.2.1 Usage Patterns and Trends

As noted, there are numerous settings where older people are likely to encounter technology including the workplace, the home, the health care arena, service, and entertainment settings. However, despite increased use of technology among older people, use of technology is still lower among older people compared with younger people. As shown in Figure 35.4, although use of the Internet among older people is increasing, it is still lower than that among younger age groups. In 2010, about 42% of people aged 65+ were Internet users compared with 78% of people aged 50–64 and 87% of those 30–49 years old. Among those 65 years and older, only about 25% of those 75–84 years of age and 5% of those 85+ years are computer or Internet users (Charness, Fox, and Mitchum 2010). Furthermore, people aged 65+ are much less likely than younger people to have a high-speed Internet connection. In 2010, 31% of adults aged 65+ in the United States had broadband access at home compared with 75% of those aged 30–49 years and 61% of those aged 50–64 years (Figure 35.5). In addition, seniors who do have Internet access and high-speed access tend to be white, highly educated, and living in households with higher incomes (Pew Internet & American Life Project 2004, 2010). Recent data (Czaja et al. 2006; Fox 2010) also indicate that older adults are less likely than younger adults to use other forms of technology such as cell phones, automatic teller machines, or DVRs. For example, data from the Pew Internet and American Life survey found that, although 82% of adults in the United States have a cell phone, only 57% of people aged 65+ years report cell phone ownership. Use of technology also tends to be lower among people with chronic conditions. According to recent data from the Pew Internet and American Life Project (Fox and Purcell 2010), living with a chronic disease has an independent negative effect on someone's likelihood to have Internet access, especially if they have more than one chronic condition. The majority of older adults have at least one chronic condition such as arthritis or hypertension and many have multiple conditions. In addition, many older people report difficulty seeing, hearing, and ambulating (Administration on Aging 2009). All these limitations can have an impact on one's ability to successfully use technology.

This age-related digital divide puts older adults, especially those in the older cohorts, at definite disadvantage in terms of their ability to live and function independently in today's technology-oriented society. It can hamper their ability to access needed information and services; compete successfully in today's labor market; engage in health care management activities; and perform routine task such as shopping, money management, and bill paying. Furthermore, the full benefits of technology may not be realized by older populations.

Technology holds great potential for improving the quality of life for older people. For example, the Internet can facilitate linkages between older adults and health care providers and communication with family members and friends, especially those who are at long distant. It is quite common within the United States for family members to

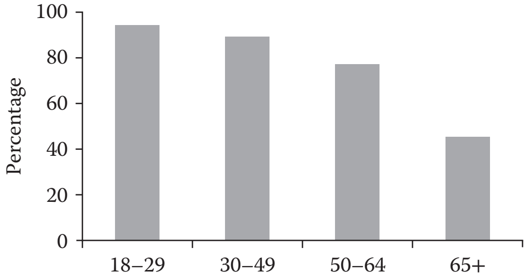


FIGURE 35.4 Percent of Internet users by age group. (From Smith, A. *Home Broadband 2010*. Pew Internet & American Life Project Surveys, 2010. http://pewinternet.org/Reports/2010/Home-Broadband-2010.aspx. With permission.)

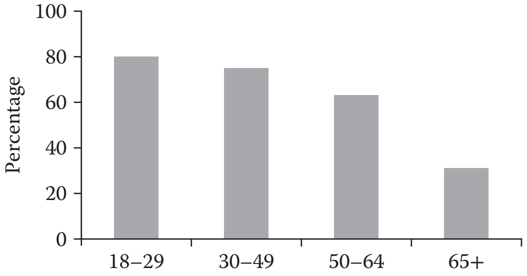


FIGURE 35.5 Percent of broadband users by age group. (Adapted from Smith, A. *Home Broadband 2010*. Pew Internet & American Life Project Surveys, 2010. http://pewinternet.org/Reports/2010/Home-Broadband-2010.aspx; U.S. Census Bureau, Current Population Survey, October 2003.)

be dispersed among different geographic regions. In fact, nearly 7 million Americans are long-distance caregivers for older relatives (Family Caregiver Alliance 2010). Clearly, network linkages can make it easier for family members to communicate with distant relatives, especially those who live in different time zones. Current technologies can also facilitate the ability of caregivers to monitor older relatives who are in need of care and support. Monitoring technologies may be especially beneficial for caregivers who are employed and must juggle work and caregiving responsibilities. A significant portion of workers are also providing care to elder family members. Recent estimates indicate that about 35% of all workers report that they are currently providing, or have recently provided, care to someone aged 65+ (Family Caregiver Alliance 2010).

The Internet may also be used to help older people communicate with health care providers. For example, telemedicine applications allow direct communication between health care providers and patients. The number of e-health applications has also grown markedly in the past several years. There are millions of websites that provide information related to health and health care activities. The Internet is also increasingly being used to access peer support among those who have a chronic condition or disease or providing care to someone who is ill. Electronic medical records (EMRs) are also becoming a fundamental component of health care. Many EMR systems include a patient portal that enables people to engage in health care management tasks, such as monitoring of laboratory/test results, medication management, appointment scheduling, and communicating with physicians and other providers.

The Internet also provides many opportunities for education. There are numerous online courses available on a myriad of topics. These opportunities will continue to expand and be more enhanced with future developments in multimedia and videoconferencing technologies. The Internet can also help older people access information about community services and resources and facilitate the performance of routine tasks such as financial management or shopping. Access to these resources and services may be particularly beneficial for older people who have mobility restrictions or lack of transportation. Finally, many government services such as Medicare and Social Security have online information and are moving toward "online" application processes. Clearly, technology is becoming an integral component of our everyday lives. The following section (35.2.2) will present a more detailed discussion of the potential use of technology by older adults. This will be followed by a discussion of the implications of aging for system design.

35.2.2 Technology Usage in Everyday Domains

35.2.2.1 Work Environments

Technology is ubiquitous within the work domain and a fundamental component of most jobs. Most workers, across all occupational sectors, need to interact with some form of technology on a daily basis. The introduction of technology into the workplace has dramatically changed work processes and the content of jobs. Workers are continually confronted with the need to adapt to new technologies, interface modifications to existing technology applications (e.g., new versions of a software application), and new ways of performing work. For many workers, existing job skills and knowledge are becoming obsolete and new knowledge and skills are required to meet the demands of today's jobs, creating enormous needs for worker training and retraining. Issues of skill obsolescence and training are especially significant for older workers as they are less likely to have had exposure to technologies such as the Internet (Czaja et al. 2006; The Pew Internet & American Life Study 2010). In addition, there is data to suggest that employers invest less in training older workers than younger workers (Barth, McNaught, and Rizzi 1993). Data from a recent focus group study that examined issues regarding barriers confronting older adults seeking employment found that the most common perceived obstacles to employment included age, insufficient qualifications, and lack of technical/computer skills. The sample included community-dwelling adults ranging in age from 51 to 76 years (Lee, Czaja, and Sharit 2009). It is important to point out that numerous studies have shown (e.g., Czaja et al. [2001]; Sharit et al. [2004]) that older adults are willing and able to learn technology-based work tasks. However, they must be provided with access to training, and training programs must be designed to accommodate the learning needs and training preferences of older learners. As will be discussed later in this chapter, much research is needed regarding the optimal training formats for older workers.

In this regard, technology is also changing the way worker training is being delivered. Technology-mediated learning or "e-learning" is quickly emerging as a preferred method for training employees. Use of e-learning tools may present special challenges for older learners who have limited technical skills or limited access to technology or broadband access. Other potential disadvantages of e-learning include the lack of face-to-face interaction with instructors or peers (Czaja and Sharit 2009). To date, there has only been limited research examining where mature learners can effectively use e-learning programs.

Technology is also influencing where work is being performed. The number of people engaging in telework is rapidly increasing. Telework encompasses a number of work arrangements, including home-based work, satellite office, and neighborhood telework centers. It can also be done on a full- or part-time basis. In 2001, about 29 million workers in the United States engaged in some form of telecommuting, and it is estimated that there could be slightly more than 40 million telecommuters by 2010 (Potter 2003). Telecommuting may be particularly appropriate for older adults, as they are more likely than younger people to be "mobility impaired" or engaged in some form of caregiving. Telecommuting also allows for more flexible work schedules and autonomy and is more amenable to part-time work. These job characteristics are generally preferred by older people. Sharit et al. (2004) found that older people are interested in pursuing this type of work as it provides them with an opportunity to remain productive and engaged in work activities at a more flexible schedule. However, more recent data (Sharit et al. 2009) from a study that examined managers' attitudes toward telework and the importance of various worker attributes they consider important to telework. Overall, the managers' attitudes were mixed with respect to potential employment of older workers as teleworkers. For example, the managers indicated that trust, maturity, time management ability, and the ability to work independently were important attributes of teleworkers. They also rated older workers high on these attributes compared with younger workers. However, the managers also perceived older workers as being inflexible, not being able to adjust to teamwork, and having less ability to keep up with changes in technology. Telework also typically involves the use of computers and the Internet, which may be problematic for older people as they are less likely to have computer skills. The prevalence of telecommuting also raises other interesting issues such as strategies to keep workers updated on changes in job demands and that can be used to provide teleworkers with technical support.

Technology will have a continuing major impact on the future structure of the labor force. Most of the job growth in the next decade will come from three occupational groups: (1) computer and mathematical occupations, (2) health care practitioners and technical occupations, and (3) education, training, and library occupations. Other occupations that will experience growth include management and financial occupations; sales and related occupations; office and administrative support operations; and installation, maintenance, and repair occupations, especially within the telecommunications industry (U.S. Bureau of Labor Statistics 2010). Internet and communication technologies will be an integral component of most of these occupations. In addition, new technology applications for job-related tasks are continually being developed.

Overall, these employment trends have significant implications for older adults. Current labor projects indicate that by 2025 the number of workers aged 55+ will be about 33 million (U.S. General Accounting Office 2003). There will also be an increase in the number of workers aged 65+. To insure that older adults are able to compete successfully in today's labor market and adapt to new workplace technologies, it is important that older adults are provided with access to retraining programs and incentives to invest in learning new skills. Greater attention also needs to be given to the design of training and instructional materials for older learners, and it is important that the abilities and characteristics of older workers are accounted for in the design of technical systems. The role of adaptive technologies in helping to make work more viable for older people especially those with some type of chronic condition or disability also needs further exploration. For example, there are a number of technologies available that can aid people with blindness or low vision such as portable Braille computers, speech synthesizers, optical character recognition, screen enlargement software, or video magnifiers. Similarly, personal amplifying devices and amplified telephone receivers can be used to aid persons with hearing loss, and voice recognition software may be beneficial for persons who have limited ability to use traditional input devices such as a mouse or keyboard because of hand or finger limitations. Clearly, there are a number of technologies that can improve the ability of older adults to function in work environments; however, the availability of these technologies does not guarantee their success. The degree to which these technologies improve the work life of older persons depends on the usability of these technologies, the availability of these technologies within organizations, the manner in which these technologies are implemented (e.g., training), and the willingness of older people to use these devices.

35.2.2.2 Home Environments

There are a number of ways older people can use computers at home to enhance their independence and quality of life. The Internet can provide access to information and services and can be used to facilitate the performance of tasks such as banking and grocery shopping. Many older people have problems performing these tasks because of functional limitations, restricted mobility, lack of transportation, and fear of crime (Nair 1989). For example, about 42% of older people report that they have functional limitations and many report that they are unable to perform tasks such as walking 2 or 3 blocks, lifting, shopping, managing money, or managing medications (Administration on Aging 2009; Federal Interagency Forum on Aging-Related Statistics 2010). Recently, a relatively large focus group study was conducted that examined use and attitudes about technology in the context of their home, work, and health care. The sample included 113 older adults, males and females, ranging in age from 65 to 85 years. The findings indicated that one of the reasons they liked technology was that use of some technologies could provide support for activities. Within home, these activities included communication (e.g., e-mail), cooking (e.g., microwaves), leisure and hobby activities, and finding information (Mitzner et al. 2010).

Several studies (e.g., Czaja et al. [1993]; Mitzner et al. [2010]; Madden [2010]) have shown that older adults are receptive to using e-mail as a form of communication and that e-mail is effective in increasing social interaction among the elderly. In fact, recent data from the Pew Internet and American Life Study (Madden 2010) indicate that among those older adults who have Internet access, 92% of those aged 50–64 and 89% of those aged 65+ use e-mail on a daily basis. Interestingly, social networking has also become prevalent among older adults with Internet access; 20% of people aged 50–64 and about 13% of those 65+ report using social networking sites on a typical day. Use of social networking sites is more prevalent among older people with high-speed Internet access.

Increased social connectivity can be beneficial for older people, especially those who are isolated or live alone. Currently, about 30.5% (11.2 million) of all non-institutionalized older persons live alone and the proportion living alone increases with advanced age. Among women aged 75 and over, for example, about 50% live alone (Administration on Aging 2009). Cody et al. (1999) found that older adults who learned to use the Internet had more positive attitudes toward aging, higher levels of perceived social support, and higher levels of connectivity with friends and relatives. A more recent study (Guilleard, Hyde, and Higgs 2007) found that use of communication technologies such as cell phones and the Internet enabled older adults to increase indirect access to family and friends living outside their neighborhood. Substantial evidence indicates that social relationships and the extent to which individuals are integrated into the community have an impact on the health of the individual (Berkman 1995). Social isolation is associated with poorer quality of life, life satisfaction, well-being, poorer health status, and distress and mental illness (Cantor and Sanderson 1999; Cobb 1976; Dykstra 1995; Ellaway, Wood, and MacIntrye 1999; Ellis and Hickie 2001).

The Internet can also be used by older people for continuing education and cognitive engagement. There are websites and software programs available on a wide variety of topics. As noted, e-learning is becoming one of the most popular forms of training within industry and in the education industry for lifelong learners (Willis 2004). There are also formal online degree programs and opportunities to be linked via videoconferencing and networking facilities to actual classrooms. The American Association for Retired Persons offers several online courses as does SeniorNet. In fact, both organizations offer several online courses and tutorials related to computer skills and use of the Internet. Currently, SeniorNet has over 40,000 members and over 240 learning centers throughout the United States (www.seniornet.com. 2005).

These learning opportunities can enable older adults to remain intellectually engaged and active, especially those who have difficulty accessing more traditional classroom-based adult education programs. Research (e.g., Baltes and Smith [1999]) clearly shows that cognitive engagement and stimulation are important for successful aging. In fact, a recent study found that the simple act of reading was associated with reduced mortality among a visually and cognitively intact sample of men (Jacobs et al. 2006). Lifelong learning is a growing interest among older people. Currently, in the United States, more than 33 million adults aged 45+ are engaged in some form of continuing education (Adler 2002).

The Internet can also be used to create "online learning communities" that bring social interaction to learning and support the learning process. An online community refers to an aggregation of people who have a shared goal, interest, need, or activity and have repeated interactions and share resources (Preece and Maloney-Krichmar 2003). The imminent availability of the next-generation Internet and interactive multimedia programming will further expand the education experiences that are available to individuals and enable information to be tailored to the specific needs and characteristics of users. This may be particularly beneficial to older adults who often learn at a slower pace than younger people and need more instructional support. A recent pilot study (Stoltz-Loike, Morrell, and Loike 2005) found that e-learning can be an effective tool for teaching older adults technology- and business-related skills. The e-learning tool evaluated in the study was customized for older adults. However, as noted by the authors, the results were based on a small sample and the e-learning methods were not compared with other traditional training methods. They also point that many of the e-learning materials that are available on the market assume a relatively sophisticated knowledge of technology and familiarity with e-learning environments. This may be disadvantageous to older people who have less knowledge of technology and less experience with computers.

As noted, currently, there is little empirical data to guide the development of these applications. In addition, almost no research has been done with older adults. This issue is especially compelling given that multimedia applications place demands on cognitive processes such as visual search, working memory, and selective attention, which are known to decline with age.

Technology may also be used to augment the memory of older people. There are many software-based applications such as e-mail calendar functions, which are intended to support memory abilities such as the ability to remember dates or appointments. Recent research (e.g., Günther et al. [2003]; Edwards et al. [2002]; Klusmann et al. [2010]) indicates that computerized cognitive training programs can be used in older people to achieve long-term improvements in important aspects of cognition. Quite recently, Tun and Lachman (2010) examined the relationship between the computer use and cognition among people aged 32–84 years. Overall, they found that frequent computer activity is associated with good cognitive function, especially executive function across adulthood. Damianakis et al. (2010) found that the use of digital video technology to produce personalized multimedia biographies helped stimulate positive social interactions between family members and individuals with Alzheimer's disease and mild cognitive impairments. However, findings regarding the beneficial use of computers and the Internet on the cognitive functions of older adults are not conclusive. Current research (e.g., Charness and Boot [2009]; McKay and Maki [2010]) is investigating the use of video-game-based training to augment the cognitive abilities of older adults.

Slegers, von Boxtel, and Jolles (2009) found that learning to use a computer and the Internet did not benefit the cognitive functioning of independent older adults. The authors discuss the fact that the differences in findings between their study and that of others may be because the participants in their study were community-dwelling elders who had less to gain from the computer intervention and because the intervention may not have sufficiently challenged the cognitive capacity of the participants. Finally, they suggest that access and use of the Internet among older adults may have an impact on other outcomes such as indices of functional limitations.

Technology also offers numerous possibilities for enhancing the safety and security of older people living at home. As noted, a large proportion of older adults, especially older women, live at home alone. Systems can be programmed to monitor home appliances, electrical, and ventilation systems, and can be linked to emergency services.

35.2.2.3 Health Care

Technology also holds the promise of improving access to health care for older people and empowering them to take an active role in health self-management. Electronic links can be established between health care professionals and older clients, providing health care providers with easy access to their patients and allowing them to conduct daily status checks or to remind patients of home health care regimes. In addition, with the rapid introduction of EMRs, many of which have patient portals, patients will also be able to communicate with health care providers via e-mail messages. There are also health websites available that allow patients and consumers to ask health care providers health-related questions. Technology may also be used to facilitate health assessment and patient monitoring. Ellis, Joo, and Gross (1991) demonstrated that older people could successfully use a computer-based health-risk assessment. There are also telemedicine applications that are being used to monitor a patient's physical functioning, such as measuring blood pressure, pulse rate, temperature, and so on. These applications offer the potential of allowing many people who are at risk for institutionalization to remain at home. The application of technology in health care delivery and services will continue as current trends in health care are toward health self-management. Individuals and their families are expected to perform a range of health care tasks and interact with a vast array of medical devices and technologies within home settings.

In this regard, the Internet is shaping and having a pronounced impact on personal health behavior. Interactive health communication or "e-health" generally refers to the interaction of an individual with an electronic device or communication technology (such as the Internet) to access or transmit health information or to receive or provide guidance and support on a health-related issue (Robinson, Eng, and Gustafson 1998). The scope of e-health applications is fairly broad but mostly encompasses searching for health information, participating in support groups, and consulting with health care professionals. Most Internet users (83%) have searched the Internet for health information (Pew Internet & American Life 2010). The majority of consumers search for information on a specific disease or medical problem, medical treatments or procedures, medications, alternative treatments or medicine, or information on providers or hospitals. Reasons for the growth of online health information seeking includes easier access by a more diverse group of users to more powerful technologies, the development of participative health care models, the growth of health information that makes it difficult for any one physician to keep pace, cost containment efforts that reduce physicians time with patients, and raising concerns about self-care and prevention (Cline and Hayes 2001). In addition, many people with health conditions or who care for someone with a health condition use social media tools to receive help from people with similar issues. In this regard, recent data from the Pew Internet and American Life Project (Fox and Purcell 2010) indicate that, although people with one or more chronic diseases are less likely to have Internet access than healthy adults, those who are Internet users are more likely to participant in online discussion groups or forums that help people with health problems.

Many consumers say that the Internet has had a significant impact on their health care behavior in terms of the way they care for themselves or for others. A recent study (Taha, Sharit, and Czaja 2009) examined health information needs among older adults and differences in perceptions and use of health information between Internet users and nonusers. The findings indicated that older people use a variety of sources to find information. Many of those who used the Internet to search for health information reported that they found empowered when they were able to bring information obtained from the Internet to their doctors and that their conversations with their physicians improved as a result of having this information. The data also indicated that there were no significant reported differences between users and nonusers in difficulty finding health information or satisfaction with the information found. However, many of the nonusers indicated that they would be willing to try using the Internet if they had some training.

However, the fact that consumers have access to e-health applications has significant implications for both patients and providers. On the positive side, access to health information can empower patients to take a more active role in the health care process. Patient empowerment can result in better informed decision making, better and more tailored treatment decisions, stronger patient–provider relationships, increased patient compliance, and better medical outcomes. On the negative side, access to this wide array of health information can overload both patient and physicians, disrupt existing relationships, and lead to poor decision making on the part of consumers. For example, one major concern within the "e-health arena" is the lack of quality control mechanisms for health information on the Internet. Currently, consumers can access information from credible scientific and institutional sources (e.g., Medline Plus) and unreviewed sources of unknown quality. Inaccurate health information could result in inappropriate treatment or cause delays in seeking health care. In fact, data indicate (El-Attar et al. 2005; Czaja, Sharit, and Nair 2008) that older adults trust health information on the Internet and generally find the Internet to be a valuable source of health information. However, data also indicated that health websites can be challenging for older adults to use (Czaja, Sharit, and Nair 2008; Taha, Sharit, and Czaja 2009). Other concerns related to the ability of nonspecialists to integrate and interpret the wealth of information that is available and the ability of health care providers to keep "pace" with their patients. Physicians increasingly report that patients come to office visits armed with information on their illness or condition and treatment options (Ferguson 1998). Results from a recent Internet user survey (Pew Internet & American Life Project 2000) also indicate that access to Internet health information has an influence on consumer decisions about seeking care, treatment choices, and their interactions with physicians. Finally, some consumers may find health information difficult to access because of design features that result in usability problems, lack of training, or limited access to technology.

The Internet may also be beneficial for family caregivers who are providing care for an older person with a chronic illness or disease such as dementia. Generally, the prevalence of chronic conditions or illnesses such as dementia, diabetes, heart disease, or stroke increases with age, and consequently, older adults (especially the "oldest old") are more likely to need some form of care or assistance. Family members are the primary and preferred source of help for elders. Currently, at least 52 million Americans are providing care for an adult who is ill or disabled (Family Caregiver Alliance 2010).

Current information technologies offer the potential of providing support and delivering services to caregivers and other family members. Networks can link caregivers to each other, health care professionals, community service, and educational programs. Information technology can also enhance a caregiver's ability to access health-related information or information regarding community resources. The Pew Internet and American Life Project (Fox 2010) queried caregivers, who said they had found the Internet to be crucial or important during a loved one's recent health crisis, about the Internet's specific role during that time. The findings indicated that the Internet helped the caregivers find advice or support from other people (36%), helped them find professional or expert services (34%), and helped them find information or compare options (26%). Recent findings from an interview study of approximately 1500 caregivers indicated that 53% of the caregivers reported that they use the Internet as a source of information about caregiving (National Alliance for Caregiving and AARP 2009).

Several studies have also shown that the computer networks (e.g., Czaja and Rubert [2002]; Gallienne, Moore, and Brennan [1993]) can increase social support for caregivers. Technology can also aid caregivers' ability to manage their own health care needs and those of the patient by giving them access information about medical problems, treatments, and prevention strategies. Software is available on several health-related topics such as stress management, caregiving strategies, and nutrition. There are also several websites available that provide information for caregivers, such as that of the Alzheimer's Association (www.alzheimers.com), National Alliance for Caregiving, and Family Caregiver Alliance. The Senior Health Website of the National Institutes of Health also contains caregiving information (http://nihseniorhealth.gov/).

Clearly, technology holds the promise of improving the quality of life for older adults and their families. However, for the full potential of technology to be realized for these populations, the needs and abilities of older adults must be considered in system design. As will be demonstrated in this chapter, older adults generally find technologies such as computers to be valuable and are receptive to using this type of technology. However, available data (e.g., Mead et al. [1997]; Czaja et al. [2001]; Sharit et al. [2003], [2009]; Czaja, Sharit, and Nair [2008]) also indicate that, although older people are generally willing and able to use technology, they typically have more problems learning to use and operate technology systems than younger adults. They also have less knowledge about potential uses of and how to access computers and other forms of technology. Morrell, Mayhorn, and Bennett (2000) found that the two primary predictors for not using the Internet among people aged 60+ years were lack of access to a computer and lack of knowledge about the Internet. Other barriers to computer and Internet access include cost, lack of technical support, usability problems, and creeping functionality (Morrell, Mayhorn, and Echt 2004; Adams, Stubbs, and Woods 2005). Older adults also tend to report lower confidence than younger people in their ability to learn to use computers (AARP 2002; Czaja et al. 2006). Before the full benefits of computer technology can be realized for older people, it is important to maximize the usefulness and usability of these technologies for this population. The following section will review characteristics of older adults that have relevance to the design of computer-based systems.

35.3 WHO ARE TODAY'S OLDER ADULTS?

In general, older Americans today are healthier, more diverse, and better educated than previous generations. Between 1970 and 2008, the percentage of older persons who had completed high school rose from 28% to 77.4% and about 20.5% had a bachelor's degree or higher. However, the percentage of minority older adults who had completed high school and obtained higher degrees was lower than white older adults (Administration on Aging 2009). As noted, higher levels of education are typically associated with technology adoption, and people who are better educated are more likely to use computers and the Internet.

On some indices, today's older adults are healthier than previous generations. The number of people aged 65+ reporting very good health and experiencing good physical functioning, such as ability to walk a mile or climb stairs, has increased in recent years. Disability rates among older people are also declining (Federal Interagency Forum on Aging-Related Statistics 2010). However, the likelihood of developing a disability increases with age, and many older people have at least one chronic condition such as arthritis or hearing and vision impairments (Figure 35.6).

As discussed in Section 35.2.1, disability among older adults has important implications for system design. People with disabilities, especially disabled elders and minorities with disabilities, are less likely to use technology such as the Internet (Fox and Purcell 2010).

Consistent with demographic changes in the U.S. population as a whole, the older population is becoming more ethnically diverse. The greatest growth will be seen among Hispanic persons, followed by non-Hispanic blacks. Currently, individuals from ethnic minority groups are less likely to own or use technologies such as computers. This

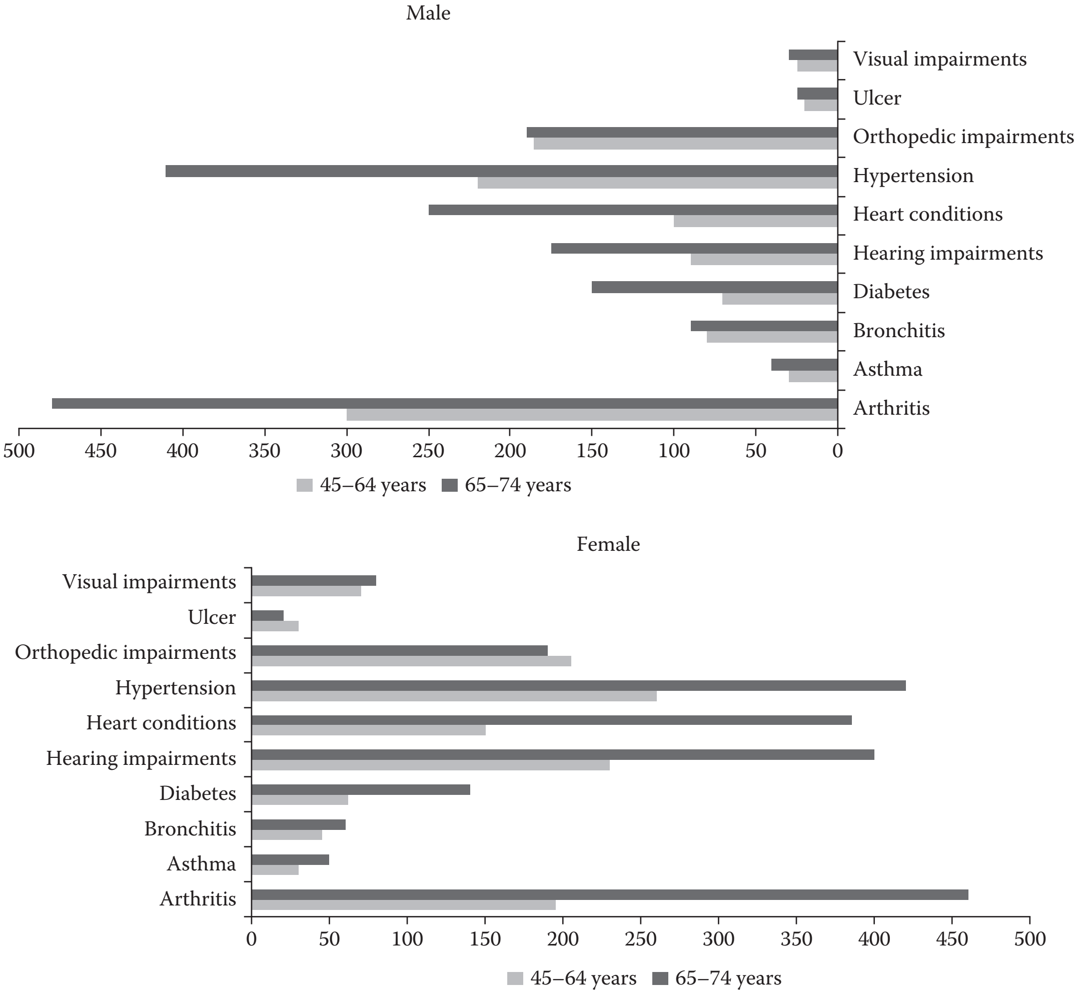


FIGURE 35.6 Chronic illness and older adults. (From Clinical Geriatrics. *Clin Geriatrics* 7(8):77. 1999. With permisson.)

implies that technology access and training programs need to be targeted for older minority populations. Also, there are gender differences in the older adult population, older women outnumber older men. In summary, health status, race, gender, educational background, cultural traditions, and economic circumstances may all influence the adoption and use of computer-based technologies. Thus, system designers need to understand the heterogeneity of the older adult population and ensure that usability testing is done with representative user groups.

35.3.1 Aging and Abilities

There are several age-related changes in functional abilities that have relevance to the design of technology systems. These include changes in sensory/perceptual processes, motor abilities, response speed, and cognitive processes. A brief review of these age-related changes in abilities is provided as a framework for understanding the potential implications of the aging process for system design. It is, however, important to recognize that there are substantial individual differences in rate and degree of functional change. Within any age group, young or old, there is significant variability in range of abilities and rate of age-related change in abilities, and this variability tends to increase with age.

35.3.2 Sensory Processes

There are a number of changes in visual abilities that have relevance to the design of computer systems. Currently, about 14 million people in the United States suffer from some type of visual impairment, and as shown in Figure 35.6, the incidence of visual impairment increases with age. Although most older people will not experience severe visual impairments, they may experience declines in eyesight sufficient to make it more difficult to perceive and comprehend visual information. This has vast implications for the design of technology systems given that many technology interfaces are primarily based on visually presented text. Visual decrements may make it more difficult for older people to perceive small icons on toolbars, read e-mail, or locate information on complex screens or websites (e.g., Charness and Holley [2001]). Age-related changes in vision also have implications for the design of instruction manuals (Fisk et al. 2009).

Aging is also associated with declines in auditory acuity. Many older adults experience some decline in auditory function such as a loss of sensitivity for high-frequency tones; difficulty understanding speech, especially if the speech is distorted; problems localizing sounds; problems in binaural listening; and increased sensitivity to loudness (Schieber et al. 1991). These changes in audition are also relevant to design of technology systems. Older people may also find it difficult to understand synthetic speech as this type of speech is typically characterized by some degree of distortion. Multimedia systems with voice output may also be problematic for older adults. High-frequency alerting sounds associated with alarm or emergency systems may also be difficult for older adults to detect.

35.3.3 Motor Skills

Aging is also associated with changes in motor skills, including slower response times, declines in ability to maintain continuous movements, disruptions in coordination, loss of flexibility, and greater variability in movement (Fisk et al. 2009). The incidence of chronic conditions such as arthritis also increases with age (Figure 35.6). These changes in motor skills have direct relevance to the ability of older people to use current input devices such as a mouse or a keyboard. For example, various aspects of mouse control such as moving, clicking, fine-positioning, and dragging are likely to be difficult for older people (Smith et al. 1999; Riviere and Thakor 1996; Walker, Millians, and Worden 1996; Charness, Bosman, and Elliott 1995).

Findings from these studies suggest that alternative input devices might be beneficial for older people (Charness and Holley 2001; Fisk et al. 2009). Murata and Iwase (2005) compared target pointing times among younger, middle-aged, and older adults using a mouse and a touch panel. They found that pointing time was longer for the older age group when using a mouse but that there were no significant age-related differences in pointing time with the touch panel. On the basis of these results, they recommend that for pointing tasks, a touch-panel interface should be used for middle-aged and older adults and provide design guidelines for touch-panel interfaces. General guidelines for the design of input devices for older adults are available (Fisk et al. 2009).

35.3.4 Cognitive Abilities

Age-related changes in cognition have important implications for the performance of technology-based tasks. It is well established that many cognitive abilities include attentional processes, working memory, discourse comprehension, problem solving and reasoning, inference formation and interpretation, and encoding and retrieval in memory decline with age. (Park 1992; Fisk et al. 2009). Aging is also associated with declines in information processing speed. Older people tend to take longer to process incoming information and typically require a longer time to respond.

35.4 WILLINGNESS AND ABILITY OF OLDER ADULTS TO ADOPT INFORMATION TECHNOLOGIES

One issue that warrants discussion when considering age and information technology is the degree to which older people are willing to interact with these types of systems and the factors that influence technology adoption. A commonly held belief is that older people are resistant to change and unwilling to interact with "high tech" products. However, the available data largely dispute this stereotype. The majority of studies that have examined the attitudes of older people toward computer technology indicate that older people are receptive to using computers. However, older people do report more computer anxiety, less computer self-efficacy, and less comfort using computers than younger adults (Nair, Lee, and Czaja 2005; Czaja et al. 2006). Furthermore, computer anxiety and computer self-efficacy are important predictors of technology adoption (Ellis and Allaire 1999; Czaja et al. 2006). However, the data also indicate that attitudes toward technology and comfort using technology are influenced by experience and the nature of interactions with computer systems and system design (e.g., Jay and Willis [1992]; Charness, Schumann, and Boritz [1992]; Dyck and Smither [1994]; Czaja and Sharit [2003]; Adams, Stubbs, and Woods [2005]). Not surprisingly, the data indicate that older adults who have a positive perception of factors such as usability, ease of use, and usefulness of technology applications and a positive interaction with technology have more positive attitudes and greater technology efficacy and are more likely to use technology applications such as the Internet.

Numerous studies (e.g., Elias et al. [1987]; Gist, Rosen, and Schwoerer [1988]; Zandri and Charness [1989]; Czaja et al. [1986]; Czaja, Hammond, and Joyce [1989]; Charness, Schuman, and Boritz [1992]; Morrell et al. [1995]; Mead et al. [1997]; White et al. [2002]) have examined if the older adults can learn to use technology such as computers and the Internet. These studies encompass a variety of technology applications and also vary with respect to training strategies. The influence of other variables, such as attitude toward computers and computer anxiety, on learning has also been examined.

Overall, the results of these studies indicate that older adults are, in fact, able to use technology such as computers and the Internet for a variety of tasks. However, they are typically slower to acquire new skills than younger adults and generally require more help and "hands-on" practice. Also, compared with younger adults on performance measures, older adults often achieve lower levels of performance. However, the literature also indicates that training interventions can be successful in terms of improving performance and it points to the importance of matching training strategies with the characteristics of the learner (Charness, Czaja, and Sharit 2007). It is also important to provide older people with training on the potential use of the technical system (e.g., what the Internet can be used for) and training on basic procedural operations (e.g., use of the mouse). As one would expect, the "usability" of the system from both a hardware and software perspective is also important (Fisk et al. 2009).

Mayhorn and colleagues (Mayhorn et al. 2004; Fisk et al. 2009) provide suggestions for development of effective computer training for older adults. Generally, these guidelines stress the importance of considering the goals, abilities, and experience levels of older adults in the design and evaluation of instructional programs and materials. Also, given the important role of anxiety and self-efficacy in technology adoption (Czaja et al. 2006), it is important that training environments are relaxed and strategies that reduce anxiety and increase self-efficacy are incorporated into training programs.

Similarly, studies have examined the ability of older people to perform technology-based tasks such as computer-based tasks that are common in work settings. For example, Czaja and colleague (1993, 1998, 1998a, 1998b, 2001; Sharit et al. 2009) conducted a series of studies examining age performance differences on a variety of simulated computer-based tasks (e.g., data entry, inventory management, customer service, and telework). Overall, the results of these studies indicate that older adults are willing and able to perform these types of tasks. However, generally the younger adults performed at higher levels than the older people. Importantly, the data also indicated that there was considerable variability in performance among the older people, and that with task experience, those in their middle years (40–59 years) performed at roughly the same levels as the young adults. In fact, task experience resulted in performance improvements for people of all ages. The results also indicated that interventions such as redesigning the screen, providing on-screen aids, and reconfiguring the timing of the computer mouse improved the performance of all participants.

Other investigators have examined age as a potential factor impacting the ability to use the Internet for information search and retrieval (e.g., Sharit et al. [2008]; Czaja, Sharit, and Nair [2008]; Czaja et al. [2010]). This is an important area of investigation given that this is one of the most common reasons people use technology such as computers. For example, this type of activity is central to use of the Internet. Also, in many work settings such as department stores, airlines, hotels, utility and health insurance companies, and educational institutions, workers are required to search through computer databases and access information to respond to customer requests. Generally, the findings from these studies indicate that while older adults are capable of performing these types of tasks, there are age-related differences in performance. Furthermore, these differences appear to be related to age differences in cognitive abilities.

For example, they examined the relationship between spatial ability, spatial memory, vocabulary skills, and age and the ability to retrieve information from a computer database that varied according to how the database was structured (e.g., hierarchical vs. linear). In general, they found that the older subjects were slower in retrieving the information than the younger adults, but there were no age-related differences in accuracy. The learning rates also differed for the two groups such that the older people were slower than the younger people. They found that the slower response on the part of the older adults was dependent on general processing speed (e.g., Westerman et al. [1995]; Freudenthal [1997]; Mead et al. [1996]; Sharit et al. [2008]; Pak et al. [2008]; Czaja et al. [2010]).

Information seeking is a complex process and places demands on cognitive abilities such as working memory, spatial memory, reasoning, and problem solving. Information seeking within electronic environment also requires special skills such as knowledge related to the search system. Given that older adults typically experience declines in cognitive abilities, such as working memory, and are less likely than younger people to have knowledge of the structure and organization of search systems, a relevant question is the degree to which they will experience difficulty searching for information in electronic environments. Generally, the available literature suggests that older adults are able to search and retrieve information within "electronic environments." However, they appear to have more difficulty than do younger adults and tend to use less efficient navigation strategies. They also appear to have problems remembering where and what they searched. To maximize the ability of older people to successfully interact with electronic information systems such as the Internet and have access to the "information highway," we need to have an understanding of the source of age-related difficulties. This type of information will allow us to develop interface design and training strategies to accommodate individual differences in performance. Currently, there is very little information on problems experienced by older people when attempting to learn and navigate the Internet, especially in real-world contexts.

35.5 DESIGNING TECHNOLOGY SYSTEMS TO ACCOMMODATE OLDER ADULTS

As discussed in Section 35.1, there are age-related changes in functioning that have implications for the design of technology systems. For example, careful attention needs to be paid to the design of display screens, placement, size, shape, and labeling of controls and design and layout of instructional materials and manuals. Design features such as character size and contrast are especially important for older computer users. Generally, larger characters and high-contrast displays are beneficial for older people. This may not be a major problem with most computers used in the home or the workplace, as it is relatively easy to enlarge screen characters. However, it may be an issue for computers in public places such as information kiosks and ATM machines. In addition to character size and contrast, it is also important to minimize the presence of screen glare (Fisk et al. 2009).

The organization and amount of information on a screen is also important as there are declines in visual search skills and selective attention with age. Only necessary information should be presented on a screen and important information should be highlighted. Further principles of perceptual organization, such as grouping, should be applied. Caution must also be exercised with respect to the use of color coding as there are declines in color discrimination abilities with age. Finally, as far as possible, information should be in a consistent location and important information should stand out and be in central locations. A flaw with many existing web pages is that advertising information has more prominence than the site information, for example, pharmaceutical advertisements on health websites.

Designing and labeling of controls and input devices also need special consideration. Although there is a growing body of research examining the relative merits and disadvantages of various input devices, there are only a few studies that have examined age effects. Generally, the existing data suggest that commonly used input devices such as keyboards and the mouse may be problematic for older people. More research needs to be directed toward identifying the efficacy of alternative input devices for older people, such as speech interfaces, especially for those who have restrictions in hand function.

Given that there are age-related changes in cognitive processes, such as working memory and selective attention, it is the interface style and usability of the interface that will have a significant influence on the performance of older adults. For example, systems should place minimal demands on cognitive abilities such as working memory, selective attention, and spatial abilities. On-screen aids such as maps and history markers may also prove to be beneficial for older people. We found, for example, that a simple graphical aid that depicted the structure of the menu system helped older adults use interactive telephone menu systems (Sharit et al. 2003).

Clearly, there are a number of interface issues that need to be investigated. At the present time, there are guidelines with respect to designing interfaces to accommodate the needs of older users (Fisk et al. 2009). The National Institute on Aging and National Library of Medicine also published guidelines for web design for older adults. In addition, the World Wide Web Consortium provides guidelines for web pages and software for persons with disabilities (http://www.w3.org/TR/WAI-WEBCONTENT/). However, as discussed by Hanson and Crayne (2005), guidelines or standards do not guarantee a good experience for all users. Usability testing with representative user groups is the cornerstone of good design. There is also an abundance of research that needs to be carried out within this area to inform the design of future systems.

In addition to general guidelines for the design of training programs and interface parameters, there are other issues that need to be considered to ensure that the benefits of technology are maximized for older people. As noted, the older adult population is very heterogeneous in terms of culture/ethnicity, health, education, and experience with technology, so it is important to take into account the wide range of abilities, needs, and desires of older users to ensure that technology can adapt to their individual differences. For example, it is important to consider different designs (e.g., languages) and approaches (e.g., advertisements) for different cultural/ethnic groups. It is also important to consider how to customize or adjust interfaces and response devices to accommodate those with varying abilities. Privacy issues are also an important concern as the use of technologies become more widespread for health and financial and service applications. For example, Demiris et al. (2004) found that privacy concerns are important to older adults with respect to the acceptance of smart home technology. Recently, Beach et al. (2009) conducted a large-scale, national survey regarding attitudes about the acceptability of using quality of life technology to gather and share information about performance of everyday activities. The sample included baby boomers and older adults with and without disabilities. Overall, the findings indicated that individuals reporting disability in the form of activity limitations had consistently more positive attitudes toward sharing information than those without disabilities. However, attitudes varied somewhat according to the nature of the information being shared, such that sharing information about activities such as toileting was less acceptable. In addition, the respondents were less positive about sharing information with the government and insurance companies than family members.

The issue of trust also needs consideration. Trust and ultimately acceptance of technology may be weakened by unreliability or excessive complexity. There are also potential problems with "over-trust," which can be serious if technology fails or if the information provided by the technology has low credibility. As noted, this is a current concern with health websites. Issues of safety and maintainability must also be addressed. Finally, the cost of systems needs to be considered not only in terms of finances but also in terms of effort required on the part of the user with respect to technology access and learning and maintenance requirements.

35.6 CONCLUSIONS

There are many areas where older people are likely to interact with technology including the workplace, the home, service, and health care settings. Current data indicate that older adults are generally receptive to using technology but often have more difficulty than younger people acquiring the skills needed to use current technology systems. They generally report more anxiety about using technologies such as computers and express less confidence in their ability to learn to interact with these systems than do younger adults. This presents a challenge for the HCI community.

For many older people, especially those who are frail, isolated, or have some type of mobility restrictions, access to technologies such as the Internet hold promise of enhancing independence and quality of life by providing linkages to goods and resources, facilitating communication, and enhancing opportunities for work and lifelong learning. Technology can also enhance the delivery of health services to older adults and help them access information on health-related topics. However, before the potential of technology is realized for older adults, the needs, preferences, skills, and abilities of older people need to be understood by system designers. As discussed by Dickinson and Gregor (2006) and Slegers, van Boxtel, and Jolles (2009), there is currently little systematic evidence to support the notion that use of computers and the Internet in and of themselves have a positive effect on the well-being of older adults. Many of the studies that have examined this issue have been plagued by methodological shortcomings. As such, there is a need for more rigorous research in this area. For example, Slegers and colleagues suggest that it is important to identify older populations who would benefit from Internet-based interventions.

Although research in this area has grown, there are many unanswered questions. For example, there are still many issues regarding design of input devices and interface design such as how to best design speech recognition systems, menus, and help systems to accommodate older people. We also know little about the efficacy of design aids and support tools for older adults. In addition, we need more information on how to best train older adults to learn to use new technologies, and there are many questions regarding the design of online training programs and multimedia formats. Issues regarding privacy and trust in technology also represent critical areas of needed research. There are also many questions related to the Internet, which remain unanswered, such as how does access to Internet information impact health care behavior and how do we best train seniors to identify and integrate the enormous amount of information that is available on the Internet? We also need to examine how technology in the workplace impacts employment opportunities and the work performance of older people. The issue of telecommuting has received little attention. In addition, we need more information on factors influencing technology adoption, especially for minority elderly or those of lower education or economic status. There are also many questions related to quality of life and socialization that need to be addressed.

Many needs of older people would be amenable to technological solutions. However, what is lacking is a systematic effort to understand these needs and incorporate them into design solutions and the marketplace (National Research Council 2004). In essence, in order to design information systems so that they are useful and usable for older people, it is important to understand the following: (1) why technology is difficult to use, when it is; (2) how to design technology for easier and effective use; and (3) how to effectively teach people to use and take advantage of technologies that are available. Answers to these questions will not only serve to benefit older adults but all potential users of technology systems.

REFERENCES

AARP. 2002. *Staying Ahead of the Curve: The AARP Work and Career Study*. Washington, DC: AARP.

Adams, N., D. Stubbs, and V. Woods. 2005. Psychological barriers to Internet usage among older adults in the UK. *Inf Health Internet Med* 30:3–17.

Adler, R. 2002. The age wave meets the technology wave: Broadband and older Americans. http://www.seniornet.org/downloads/broadband.pdf (accessed August 25, 2010).

Administration on Aging. 2009. *A Profile of Older Americans*. http://www.aoa.gov/aoaroot/aging_statistics/Profile/2009/docs/2009profile_508.pdf (accessed July 15, 2010).

Baltes, P. B., and J. Smith. 1999. Multilevel and systemic analyses of old age: Theoretical and empirical evidence for a fourth age. In *Handbook of Theories of Aging*, ed. V. L. Bengtson and K. W. Schaie, 153–73. New York: Springer.

Barth, M. C., W. mcNaught, and P. Rizzi. 1993. Corporations and the aging workforce. In *Building the Competitive Workforce*, ed. P. H. Mirvis, 156–200. New York: Wiley.

Beach, S., R. Schulz, J. Downs, J. Matthews, B. Barron, and K. Seelman. 2009. Disability, age, and information privacy attitudes in quality of life technology applications: Results from a national web study. *ACM Transact Accessible Comput* (2):1–21.

Berkman, L. F. 1995. The role of social isolations in health promotion. *Psychosomatic Med* 57:245–54.

Birren, J. E., and K. W. Schaie. 2001. *Handbook of the Psychology and Aging*. San Diego, CA: Academic Press.

Cantor, N., and C. Sanderson. 1999. Life task participation and well-being: The importance of taking part in daily life. In *Well-being: The Foundation of Hedonic Psychology*, ed. D. Kahneman, E. Diener, and N. Schward, 230–43. New York: Russell Sage Foundation.

Charness, N., and W. R. Boot. 2009. Aging and information technology use: Potential and pitfalls. *Curr Directions Psychol Sci* 18:253–8.

Charness, N., E. A. Bosman, and R. G. Elliot. 1995. Senior-friendly input devices: Is the pen mightier than the mouse? In *Paper Presented at the 103 Annual Convention of the American Psychological Association Meeting*, New York.

Charness, N., S. J. Czaja, and J. Sharit. 2007. Age and technology for work. In *Aging and Work in the 21st Century*, ed. K. S. Shultz and G. A. Adams, 225–49. Mahwah, NJ: Erlbaum.

Charness, N., M. C. Fox, and A. L. Mitchum. 2010. Lifespan cognition and information technology. In *Handbook of Lifespan Psychology*, ed. K. Fingerman, C. Berg, T. Antonnuci, and J. Smith, New York: Springer.

Charness, N., and P. Holley. 2001. Minimizing computer performance deficits via input devices and training. In *Presentation Prepared for the Workshop on Aging and Disabilities in the Information Age*. Baltimore, MD, John Hopkins University.

Charness, N., C. E. Schumann, and G. A. Boritz. 1992. Training older adults in word processing: Effects of age, training technique and computer anxiety. *Int J Aging Technol* 5:79–106.

Cline, R. J. W., and K. H. Hayes. 2001. Consumer health information seeking on the Internet: The state of the art. *Health Educ Res* 16:671–92.

Clinical Geriatrics. 1999. Trend watch: Chronic illness and the aging U.S. population. *Clin Geriactrics* 7(8):77.

Cobb, S. 1976. Social support as a moderator of life stress. *Psychosomatic Med* 38:300–14.

Cody, M. J., D. Dunn, S. Hoppin, and P. Wendt. 1999. Silver surfers: Training and evaluating Internet use among older adult learners. *Commun Educ* 48:269–86.

Czaja, S. J., N. Charness, A. D. Fisk, C. Hertzog, S. N. Nair, W. Rogers, and J. Sharit. 2006. Factors predicting the use of technology: Findings from the center for research and education on aging and technology enhancement (CREATE). *Psychol Aging* 21:333–52.

Czaja, S. J., J. H. Guerrier, S. N. Nair, and T. K. Laudauer. 1993. Computer communication as an aid to independence for older adults. *Behav Inf Technol* 12:197–207.

Czaja, S. J., K. Hammond, J. Blascovich, and H. Swede. 1986. Age-related differences in learning to use a text-editing system. *Behav Inf Technol* 8:309–19.

Czaja, S. J., K. Hammond, and J. B. Joyce. 1989. Word processing training for older adults. Final report submitted to the National Institute on Aging (Grant # 5 R4 AGO4647-03).

Czaja, S. J., and C. C. Lee. 2007. Information technology and older adults. In *The Human Computer-Interaction Handbook*, 2nd ed., ed. J. A. Jacko and A. Sears, 777–92. New York: Lawrence Erlbaum Associates.

Czaja, S. J., and M. Rubert. 2002. Telecommunications technology as an aid to family caregivers of persons with dementia. *Psychosomatic Med* 64:469–76.

Czaja, S. J., and J. Sharit. 1993. Age differences in the performance of computer-based work as a function of pacing and task complexity, *Psychol Aging* 8:59–67.

Czaja, S. J., and J. Sharit. 1998a. Ability-performance relationships as a function of age and task experience for a data entry task. *J Exp Psychol Appl* 4:332–51.

Czaja, S. J., and J. Sharit. 1998b. Age differences in attitudes towards computers: The influence of task characteristics. *J Gerontol Psychol Sci Social Sci* 53B:329–40.

Czaja, S. J., and J. Sharit. 2003. Practically relevant research: Capturing real-world tasks, environments, and outcomes. *Gerontologists* 43:9–18.

Czaja, S. J., and J. Sharit. 2009. Aging and Work: Assessment and implications for the future. Johns Hopkins University Press.

Czaja, S. J., J. Sharit, M. A. Hernandez, S. N. Nair, and D. Loewenstein. 2010. Variability among older adults in Internet health information–seeking performance. *Gerontechnology* 9:46–55.

Czaja, S. J., J. Sharit, and S. N. Nair. 2008. Usability of the medicare health web site. *JAMA* 300(7):790–1.

Czaja, S. J., J. Sharit, S. Nair, and M. Rubert. 1998. Understanding sources of user variability in computer-based data entry performance. *Behav Inf Technol* 19:282–93.

Czaja, S. J., J. Sharit, D. Ownby, D. Roth, and S. N. Nair. 2001. Examining age differences in performance of a complex information search and retrieval task. *Psychol Aging* 16:564–79.

Damianakis, T., M. Crete-Nishihata, K. L. Smith, R. M. Baeker, and E. Marziali. 2010. The psychosocial impacts of multimedia biographies on persons with cognitive impairments. *Gerontologist* 50:23–35.

Demiris, G., M. J. Rantz, M. A. Aud, K. D. Marek, H. W. Tyres, M. Skubic, and A. A. Hussam. 2004. Older adults' attitude towards and perceptions of 'smart home' technologies: A pilot study. *Med Inf Internet Med* 29:87–94.

Dickinson A., and P. Gregor. 2006. Computer has no demonstrated impact on the well-being of older adults. *Int J Hum Comput Stud* 64:744–53.

Dyck, J. L., and J. A. Smither. 1994. Age differences in computer anxiety: The role of computer experience, gender and education. *J Educ Comput Res* 10:239–48.

Dykstra, P. 1995. Loneliness among the never and formerly married: The importance of supportive friendships and a desire for independence. *J Gerontol Psychol Sci Social Sci* 50B:S321–9.

Edwards, J. E., V. G. Wadley, R. S. Myers, D. K. Roenker, G. M. Cissel, and K. Ball. 2002. Transfer of a speed of processing intervention to near and far cognitive functions. *Gerontol* 48:329–40.

El-Attar, T. E., J. Gray, S. Nair, R. Ownby, and S. J. Czaja. 2005. Older adults and internet health information seeking. In *Proceedings of the 49th Annual Meeting of the Human Factors and Ergonomics Society*, 163–6. Santa Monica, CA: Human Factors and Ergonomics Society.

Elias, P. K., M. F. Elias, M. A. Robbins, and P. Gage. 1987. Acquisition of word-processing skills by younger, middle-aged, and older adults. *Psychol Aging* 2:340–8.

Ellaway, A., S. Wood, and S. MacIntyire. 1999. Some to talk to? The role of loneliness as a factor in the frequency of GP consultations. *Br J Gen Pract* 49:363–637.

Ellis, E. R., and A. J. Allaire. 1999. Modeling computer interest in older adults: The role of age, education, computer knowledge, and computer anxiety. *Hum Factors* 41:345–55.

Ellis, P., and I. Hickie. 2001. What causes mental illness. In *Foundations of Clinical Psychiatry*, ed. S. Bloch and B. Singh, 43–62. Melbourne: Melbourne University Press.

Ellis, L. B. M., H. Joo, and C. R. Gross. 1991. Use of a computer-based health risk appraisal by older adults. *J Family Pract* 33:390–4.

Family Caregiver Alliance. 2010. Caregiving: A universal occupation. http://www.caregiver.org/caregiver/jsp/content_node.jsp?nodeid=2313 (accessed September 5, 2010).

Federal Interagency Forum on Aging-Related Statistics. 2010. *Older Americans 2010: Key Indicators of Well-being*. Washington, DC: U.S. Government Printing Office.

Ferguson, T. 1998. Digital doctoring: Opportunities and challenges in electronic-patient communication. *J Am Med Assoc* 280:1261–2.

Fisk, A. D., W. A. Rogers, N. Charness, S. J. Czaja, and J. Sharit. 2009. Designing for older adults: Principles and creative human factors approaches, 2nd ed. Boca Raton, FL: CRC Press.

Fox, S. 2010. *Health Pew Internet & American Life Project*. http://www.pewinternet.org/topics/health.aspx (accessed August 30, 2010).

Fox, S., and K. Purcell. 2010. Chron disease and the Internet. *Pew Internet Am Life Project* http://www.pewinternet.org/Reports/2010/Chronic-Disease.aspx (accessed August 30, 2010).

Freudenthal, D. 1997. *Learning to use Interactive Devices; Age Differences in the Reasoning Process*. Master's thesis, Eindhoven University of Technology.

Gallienne, R. L., S. M. Moore, and P. F. Brennan. 1993. Alzheimer's caregivers: Psychosocial support via computer networks. *J Gerontol Nursing* 12:1–22.

Gist, M., B. Rosen, and C. Schwoerer. 1988. The influence of training method and trainee age on the acquisition of computer skills. *Pers Psychol* 41:255–65.

Guilleard, C., M. Hyde, and P. Higgs. 2007. The impact of age, place, aging in place, and attachment to place on the well-being of the over 50s in England. *Res Aging* 29:590–605.

Günther, V. K., P. Schäfer, B. J. Holzner, and G. W. Kemmler. 2003. Long-term improvement in cognitive performance through computer-assisted cognitive training: A pilot study in a residential home for older people. *Aging Ment Health* 7(3):200–06.

Jacobs, J. M., R. Hammerman-Rozenberg, A. Cohen, J. Stressman. 2006. Chronic back pain among the elderly: Prevalence, associations, and predictors. *Spine* 31:E203–7.

Jay, G. M., and S. L. Willis. 1992. Influence of direct computer experience on older adults attitude towards computer. *J Gerontol Psychol Sci* 47:250–7.

Klusmann, V., A. Evers, R. Schwarzer, P. Schlattmann, F. M. Reischies, I. Heuser, and F. C. Dimeo. 2010. Complex mental and physical activity in older women and cognitive performance: A 6-month randomized controlled trial. *J Gerontol Series A* 65A:680–8.

Lee, C. C., S. J. Czaja, and J. Sharit. 2009. Training older workers for technology-based employment. *Educ Gerotology* 35:15–31.

Madden, M. 2010. Older adults and social media. Pew Internet & American Life Project. http://pewinternet.org/Reports/2010/Older-Adults-and-Social-Media.aspx (accessed August 30, 2010).

Mayhorn, C. B., A. J. Stronge, A. C. McLaughlin, and W. A. Rogers. 2004. Older adults, computer training, and the systems approach: A formula for success. *Educ Gerontol* 30:185–204.

McKay, S. M., and B. E. Maki. 2010. Attitudes of older adults towards shooter video games: An initial study to select an acceptable game for training visual processing. *Gerontechnology* (9):5–17.

Mead, S. E., R. A. Sit, B. A. Jamieson, G. K. Rousseau, and W. A. Rogers. 1996. On-line library catalog: Age-related differences in performance for novice users. Paper presented at the Annual Meeting of the American Psychological Association, Toronto, Canada.

Mead, S. E., V. A. Spaulding, R. A. Sit, B. Meyer, and N. Walker. 1997. Effects of age and training on World Wide Web navigation strategies. In *Proceedings of the Human Factors and Ergonomics Society 41st Annual Meeting*, 152–6. Santa Monica, CA: Human Factors and Ergonomics Society.

Mitzner, T. L., J. B. Boron, C. B. Fausset, A. E. Adams, N. Charness, S. J. Czaja, K. Dijkstra, A. D. Fisk, W. A. Rogers, and J. Sharit. 2010. Older adults talk technology: Technology use and attitude. *Comput Hum Behav* 26:1710–21.

Morrell, R. W., C. B. Mayhorn, and J. Bennett. 2000. A survey of World Wide Web in middle-aged and older adults. *Hum Factors* 42(2):175–85.

Morrell, R. W., C. B. Mayhorn, and K. V. Echt. 2004. Why older adults use or do not use the Internet. In *Gerontechnology: Research and Practice in Technology and Aging*, ed. D. C. Burdkick and K. Kwon, 86–96. New York: Springer.

Morrell, R. W., D. C. Park, C. B. Mayhorn, and K. V. Echt. 1995. Older adults and electronic communication networks: Learning to use ELDERCOMM. Paper presented at the 103 Annual Convention of the American Psychological Association. New York, New York.

Murata, A., and H. Iwase. 2005. Usability of touch-panel interfaces for older adults. *Hum Factors* 47(4):767–76.

Nair, S. N. 1989. *A Capability-Demand Analysis of Grocery Shopping Problems Encountered by Older Adults*. A thesis design submitted to the department of Industrial Engineering, State University of New York at Buffalo in partial fulfillment for the requirements for Master of Science.

Nair, S. N., C. C. Lee, and S. J. Czaja. 2005. Older adults and attitudes toward computers: Have they changed with recent advances in technology? In *Proceedings of the 49th Annual Meeting of the Human Factors and Ergonomics Society*, 154–7. Santa Monica, CA: Human Factors and Ergonomics Society.

National Alliance for Caregiving and AARP. 2009. Caregiving in the U.S. 2009. http://www.aarp.org/relationships/caregiving/info-12-2009/caregiving_09.html (accessed August 15, 2010).

National Center for Health Statistics. 2005. *Health, United States, 2005 with Chartbook on Trends in the Health of Americans*. Washington, DC. Hyattsville, Maryland: US Government Printing Office.

National Research Council. 2004. Technology for adaptive aging. Steering committee for the workshop on technology for adaptive aging. In *Board on Behavioral, Cognitive, and Sensory Sciences, Division of Behavioral and Social Sciences and Education*, ed. R. W. Pew and S. B. Van Hemel. Washington, DC: The National Academies Press.

Pak, R., J. Sharit, S. J. Czaja, W. A. Rogers, and A. D. Fisk. 2008. The role of spatial abilities and age in performance in an auditory computer navigation task. *Comput Hum Behav* 24:3045–51. PMID: 18997876.

Park, D. C. 1992. Applied cognitive aging research. In *The Handbook of Aging and Cognition*, ed. F. I. M. Crail and T. A. Salthouse, 449–94. New Jersey: Laurence Erlbaum Associates Pub.

Pew Internet & American Life Project. 2000. The online health care evolution: How the web helps Americans take better care of themselves. http://www.pewinternet.org/pdfs/PIP_Health_Report.pdf (accessed August 30, 2010).

Pew Internet & American Life Project. 2004. *Older Americans and the Internet*. http://www.pewinternet.org/pdfs/PIP_Seniors_Online_2004.pdf (accessed August 30, 2010).

Potter, E. E. 2003. Telecommuting: The future of work, corporate culture, and American society. *J Labor Res* XXIV:73–84.

Preece, J., and D. Maloney-Krichmar. 2003. Online communities: Focusing on sociability and usability. In *The Human Computer-Interaction Handbook*, ed. J. A. Jacko and A. Sears, 596–620. Mahwah, NJ: Lawrence Erlbaum Assoc.

Riviere, C. N., and N. V. Thakor. 1996. Effects of age and disability on tracking tasks with a computer mouse: Accuracy and linearity. *J Rehabil Res Dev* 33(1):6–15.

Robinson, T. N., P. K. Eng, and D. Gustafson. 1998. An evidence-based approach to interactive health communication: A challenge to medicine in the information age. *J Am Med Assoc* 280:1264–9.

Schieber, F., J. L. Fozard, S. Gordon-Salant, and J. W. Weiffenbach. 1991. Optimizing sensation and perception in older adults. *Int J Ind Ergonom* 7:133–62.

Schulz, R., L. S. Noelker, K. Rockwood, and R. L. Sprott. 2006. *The Encyclopedia of Aging*. Vol 1, New York: Springer.

Sharit, J., S. J. Czaja, M. Hernandez, Y. Yang, D. Perdomo, J. Lewis, C. C. Lee, and S. N. Nair. 2004. An evaluation of performance by older persons on a simulated telecommuting task. *J Gerontol Psychol Sci* 59B, No.6, P305–16.

Sharit, J., S. J. Czaja, S. N. Nair, and C. C. Lee. 2003. The effects of age and environmental support in using telephone voice menu systems. *Hum Factors* 45:234–51.

Sharit, J., M. Hernandez, S. J. Czaja, and S. N. Nair. 2009. The employability of older workers as teleworkers: an appraisal of issues and an empirical study. *Hum Factors Ergon Man* 19:457–77.

Sharit, J., M. Hernandez, S. J. Czaja, and P. Pirolli. 2008. Investigating the roles of knowledge and cognitive abilities in older adult information seeking on the web. *ACM Trans Comput Hum Interact* 15(3):1–25.

Slegers, K., M. P. J. van Boxtel, and J. Jolles. 2009. The efficiency of using everyday technological devices by older adults: The role of cognitive functioning. *Ageing Soc* 29:309–25.

Smith, A. 2010. Home Broadband 2010 Pew Internet & American Life Project Surveys. http://pewinternet.org/~/media//Files/Reports/2010/Home%20broadband%202010.pdf (accessed August 30, 2010).

Smith, N. W., J. Sharit, and S. J. Czaja. 1999. Aging, motor control, and performance of computer mouse tasks. *Hum Factors* 41(3):389–96.

Stoltz-Loike, M., R. W. Morrell, and J. D. Loike. 2005. Can e-learning be used as an effective training method for people over age 50? A pilot study. *Gerontechnol* 4(2):101–13.

Taha, J., S. J. Czaja, and J. Sharit. 2009. Use of and satisfaction with sources of health information among older Internet users and non-users. *Gerontologist* 49:663–73. PMID: 19741112.

Tun, P. A., and M. E. Lachman. 2010. The association between computer use and cognition across adulthood: Use it so you won't lose it? *Psychol Aging* 25:560–8.

U.S. Bureau of Labor Statistics. 2010. Occupations with the largest job growth. http://www.bls.gov/oco/cg/cgs035.htm.

U.S. General Accounting Office. 2003. *Older Workers: Policies of other Nations to Increase Labor Force Participation*. http://www.gao.gov/new.items/d03307.pdf.

Walker, N., J. Millians, and A. Worden. 1996. Mouse accelerations and performance of older computer users. In *Proceedings of Human Factors and Ergonomics Society 40th Annual Meeting*, 151–4. Santa Monica, CA: Human Factors and Ergonomics Society.

Westerman, S. J., D. R. Davies, A. I. Glendon, R. B. Stammer, and G. Matthews. 1995. Age and cognitive ability as predictors of computerized information retrieval. *Behav Inf Technol* 14:313–26.

White, H., E. McConnell, L. G. Branch, R. Sloane, C. Pieper, and T. L. Box. 2002. A randomized controlled trial of the psychosocial impact of providing Internet training and access to older adults. *Aging Men Health* 6(3):213–22.

Willis, S. 2004. Technology and learning in current and future older cohorts. In *Technology for Adaptive Aging*, *Board on Behavioral, Cognitive, and Sensory Sciences, Division of Behavioral and Social Sciences and Education*, ed. R. W. Pew and S. B. Van Hemel, 209–29. Washington, DC: The National Academies Press.

Zandri, E., and N. Charness. 1989. Training older and younger adults to use software. *Educ Gerontol* 15:615–31.

36 Human–Computer Interaction for Kids

Amy Bruckman, Alisa Bandlow, Jill Dimond, and Andrea Forte

CONTENTS

36.1 DESIGNING FOR AND WITH CHILDREN

How is designing computer software and hardware for kids different from designing for adults? Many researchers have addressed questions about the impact of technology on children; less has been said about the impact children can have on the design of technology. Methods for designing with and for children are only recently becoming widespread features of the design literature (see Jensen and Skov [2005]).

In designing for children, people tend to assume that kids are creative, intelligent, and capable of great things if they are given good tools and support. If children cannot or do not care to use technologies we have designed, it is our failure as designers. These assumptions are constructive, because users generally rise to designers' expectations. In fact, the same assumptions are useful in designing for adults. Designers of software for children start out at an advantage, because they tend to believe in their users. However, they may also be at a disadvantage, because they no longer remember the physical and cognitive differences of being a child.

In this chapter, we will

- Describe how children's abilities change with age, as it relates to human–computer interaction (HCI);
- Discuss how children differ from adults cognitively and physically, for those characteristics most relevant for HCI
- Discuss children as participants in the design process
- Review recommendations for usability testing with kids
- Review genres of computer technology for kids and design recommendations for each genre

36.2 HOW ARE CHILDREN DIFFERENT?

As people develop from infants to adults, their physical and cognitive abilities increase over time (Kail 1991; Miller and Vernon 1997; Thomas 1980). The Swiss psychologist Jean Piaget was a leading figure in analyzing how children's cognition evolves (Piaget 1970, pp. 29–33). Piaget showed that children do not just lack knowledge and experience, but also fundamentally experience and understand the world differently than adults. He divided children's development into a series of stages, as follows:

- Sensori-motor (birth—2 years)
- Pre-operational (ages 2–7)
- Concrete operational (ages 7–11)
- Formal operational (ages 11 and up)

Contemporary research recognizes that all children develop differently, and individuals may differ substantially from this typical picture (Schneider 1996). However, this general characterization remains useful.

In the sensorimotor stage, children's cognition is heavily dependent on what their senses immediately perceive. Software for children this young is difficult to design. Little interaction can be expected from the child. Obviously, all instruction must be given in audio, video, or animation, since babies cannot read. Furthermore, babies generally cannot be expected to use standard input devices like a mouse effectively, even with large targets.

In order to address this issue, AlphaBaby, is open-source software that allows infants and toddlers to play with the computer without inflicting harm to software. Every time a key or mouse is clicked, letters and shapes appear while sounds play (AlphaBaby 2009). Another example, "Reader Rabbit Toddler" by The Learning Company is targeted at children ages one to three. To eliminate the need for mouse clicking, the cursor is transformed into a big yellow star with room for five small stars inside it. As the mouse is held over a target, the small stars appear one at a time. When the fifth star appears, this counts as clicking on that target. If the child does click, the process simply moves faster. The only downside is the occasional unintended click on the "go back to the main menu" icon.

In most activities in Reader Rabbit Toddler, nearly random mouse movement will successfully complete the activity. For example, in the "bubble castle" activity, the child needs to rescue animals trapped in soap bubbles that are bouncing around the screen. Random mouse movements will catch the animals relatively quickly. Yet the parent or teacher watching a child's use of the software over time will typically begin to detect patterns in that mouse movement that become more and more obviously intentional—the mouse moves more and more directly toward the bubbles with animals in them. This is a particularly well-thought-out interface, because it mimics how young children learn language. A baby's first attempts at sounds are greeted with great enthusiasm—the child says an unrecognizable phoneme and the parents smile and say "You said Dada! This is Dada!" Over time, the utterance begins to really sound like the child said "Dada." An initial positive reinforcement for even the most remote attempt at the target behavior puts the child on a good learning trajectory to acquiring that behavior (Holdaway 1979).

Many examples of software and other cultural artifacts for young children are designed in accordance with adult expectations of what a child should like. There are a few noteworthy exceptions—for example, the television show Teletubbies is out of harmony with those stereotypes. Many adults find the television show bizarre and grating, but it is wildly popular with toddlers. The designers of the original BBC television series, Anne Wood and Andy Davenport, used detailed observations of young children's play and speech in their design. Wood comments, "Our ideas always come from children. If you make something for children, the first question you must ask yourself is, 'What does the world look like to children?' Their perception of the world is very different to that of grown-ups. We spend a lot of time watching very around them; what they say" (Davenport and Wood 1997). Focus groups also played an important role (BBC 1997). Young children are so radically different from adults that innovative design requires careful fieldwork.

While toddlers' interaction with software on a standard desktop computer affords limited possibilities, specialized hardware can expand the richness and complexity of interactions. For example, "Music Blocks" by Neurosmith is recommended for ages 2 and above (Figure 36.1). Five blocks fit in slots in the top of a device rather like a "boom box" portable music player. Each block represents a phrase of music. Each side of the block is a different instrumentation of that musical phrase. Rearranging the blocks changes the music (http://www.neurosmith.com). Interaction of this complexity would be impossible for 2-year olds using a screen-based interface but is quite easy with specialized hardware. Research on alternative computer interfaces such as tangible technologies (Ishii and Ullmer 1997; Dourish 2001) holds great promise for novel children's interface designs (Price et al. 2003; Rogers et al. 2004).

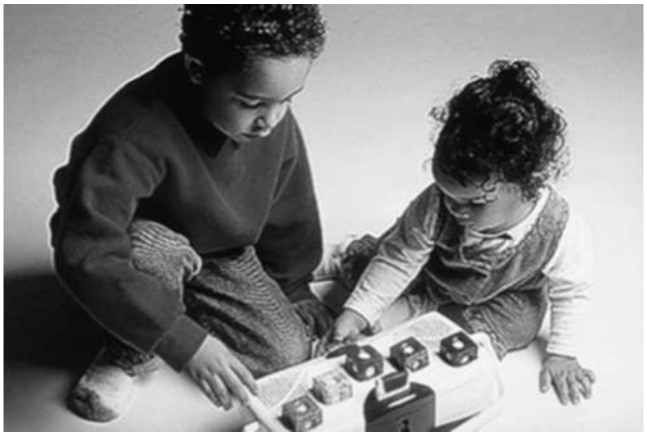


FIGURE 36.1 Children playing with Music Blocks.

In the preoperational stage (ages 2–7), children's attention span is brief. They can only hold one thing in memory at a time. They have difficulty with abstractions. They cannot understand situations from another person's point of view, which may present difficulties in partnering with other children to design technologies. While some children may begin to read at a young age, designs for this age group generally assume the children are still preliterate. It is reasonable to expect children at this age can click on specific mouse targets, but they must be relatively large. Use of the keyboard is still generally avoided by most designers (except "hit any key" approaches).

In the concrete operational stage (ages 7–11), "we see children maturing on the brink of adult cognitive abilities. Though they cannot formulate hypothesis, and though abstract concepts such as ranges of numbers are often still difficult, they are able to group like items and categorize" (Schneider 1996). Concrete operational children are old enough to use relatively sophisticated software, but young enough to still appreciate a playful approach. It is reasonable to expect simple keyboard use. Children's ability to learn to type grows throughout this age group. It is reasonable to expect relatively fine control of the mouse.

Finally, by the time a child reaches the formal operational stage (ages 12 and above), designers can assume the child's thinking is generally similar to that of adults. Their interests and tastes, of course, remain different. Designing for this age group is much less challenging, because adult designers can at least partially rely on their own intuitions.

Using age as a guide can be useful; however, designers should be aware that designing "too young" can be just as problematic as designing "too old." Children are acutely aware of their own abilities; being asked to interact with technology designed for younger children can be perceived as an affront or boring (Halgren, Fernandes, and Thomas 1995; Gilutz and Nielsen 2002). In addition to considering the implications of cognitive development on children's ability to use technologies, it is important to remember that different age groups differ culturally too. Understanding what is fun or interesting for a particular age group involves understanding both developmental ability and children's culturally dependent esthetic sensibilities. Oosterholt, Kusano, and Vries (1996) suggest that designers should avoid trying to be fashionable—what is "cool" changes quickly—and should target a limited age range because children's abilities and sensibilities change quickly as well.

In Sections 36.2.1 to 36.2.5, we will focus on several characteristics of children that are relevant for HCI research, such as the following:

- Dexterity
- Speech
- Reading
- Background knowledge
- Interaction style

36.2.1 Dexterity

Young children's fine motor control is not equal to that of adults (Thomas 1980) and they are physically smaller. Devices designed for adults may be difficult for children to use. Joiner et al. (1998) note that "the limited amount of research on children has mainly assessed the performance of children at different ages and with different input devices."

Numerous studies confirm that children's performance with mice and other input devices increases with age (Joiner et al. 1998; Hourcade 2002). Compared to adults, children have difficulty holding down the mouse button for extended periods and have difficulty performing a dragging motion (Strommen 1994). This means that many standard desktop interface features pose problems for young users. For example, kids have difficulty with marquee selection. Marquee selection is a technique for selecting several objects at once using a dynamic selection shape. In traditional marquee selection, the first click on the screen is the initial, static corner of the selection shape (typically a rectangle). Dragging the mouse controls the diagonally opposite corner of the shape, allowing you to change the dimensions of the selected area to encapsulate the necessary objects. Dragging the mouse away from the initial static corner increases the size of the selection rectangle, while dragging the mouse toward the initial static corner decreases the size of the selection rectangle. A badly placed initial corner can make it difficult and sometimes impossible to select/encapsulate all of the objects.

Berkovitz (1994) experimented with a new encirclement technique: the initial area of selection is specified with an encircling gesture and moving the mouse outside of the area enlarges it. Hourcade developed PointAssist (Hourcade, Perry, and Sharma 2008), which uses sub movements around targets to trigger a precision pointing mechanism. This software enabled 4-year olds to achieve accuracy rates similar to 18- to 22-year olds.

Kids may have trouble double-clicking, and their small hands may have trouble using a three-button mouse (Bederson et al. 1996). As with adults, point-and-click interfaces are easier to use than drag-and-drop (Inkpen 2001; Joiner et al. 1998). Inkpen (2001) notes that "Despite this knowledge, children's software is often implemented to utilize a drag-and-drop interaction style. Bringing solid research and strong results […] to the forefront may help make designers of children's software think more about the implications of their design choices."

Strommen (1998) notes that since young children cannot reliably tell their left from their right, interfaces for kids should not rely on that distinction. In his Actimates interactive plush toy designs, the toys' left and right legs, hands, and eyes always perform identical functions. More recent interactive soft toys like "Learning Baby Tad" by Leapfrog (http://www.leapfrog.com) relies on clear visual markings on left and right paws to indicate their distinct functions.

Surprisingly, there is little work that evaluates direct pointing technologies such as styluses and touch screens with children. One exception looks at designing mobile phone applications for children. Mobile phone interaction presents some challenges—moving from large screens and mouse-based interactions to small screens and touch-based interactions. Revelle and Reardon evaluate kid's interactions on a touch mobile phone application (Revelle and Reardon 2009). They recommend making the touch screen hot spots bigger and tilting features should be at a higher threshold as kids unintentionally tilt the phones more than adults.

Tangible devices also hold some promise in comparison to pointing devices for direct manipulation tasks. Antle et al. evaluated a mouse-based versus tangible-based input for a jigsaw puzzle with 132 children. They found that children were faster and more successful with the tangible-based approach (Antle, Droumeva, and Ha 2009).

36.2.2 Speech

Speech recognition has intriguing potential for a wide-variety of applications for children. O'Hare and McTear (1999) studied use of a dictation program by 12-year olds and found that they could generate text more quickly and accurately than by typing. They note that dictation automatically avoids some of the errors children would otherwise make, because the recognizer generates correct spelling and capitalization. This is desirable in applications where generating correct text is the goal. If instead the goal is to teach children to write correctly (and, for example, to capitalize their sentences), then dictation software may be counter-productive.

While O'Hare and McTear (1999) were able to use a standard dictation program with 12-year olds, Nix, Fairweather, and Adams (1998) note that speech recognition developed for adults will not work with very young children. In their research on a reading tutor for children 5 to 7 years old, they first tried a speech recognizer designed for adults. The recognition rate was only 75%, resulting in a frustrating experience for their subjects. Creating a new acoustic model from the speech of children in the target age range, they were able to achieve an error rate of less than 5%. Further gains were possible by explicitly accounting for common mispronunciations and children's tendency to respond to questions with multiple words where adults would typically provide a one-word answer. Even with the improved acoustic model, the recognizer still made mistakes. To avoid frustrating the children with incorrect feedback, they chose to have the system never tell the child they were wrong. When the system detects what it believes to be a wrong answer, it simply gives the child an easier problem to attempt.

36.2.3 Reading

The written word is a central vehicle for communicating information to humans in human–computer interfaces. Consequently, designing computer technology for children with developing reading skills presents a challenge. Words must be chosen that are at an appropriate reading level for the target population. Larger font sizes are generally preferred. Bernard et al. (2001) found that kids 9 to 11 years old prefer 14-point fonts over 12-point. Surprisingly, very little empirical work has been done in this area. Most designers follow the rule of thumb that the younger the child, the larger the font should be.

Designing for pre-literate children presents a special challenge. Audio, graphics, and animation must substitute for all functions that would otherwise be communicated in writing. The higher production values required can add significantly to development time and cost.

36.2.4 Background Knowledge

Many user interfaces are based on metaphors (Erickson 1990) from the adult world. Jones (1992) notes that children are less likely to be familiar with office concepts like file folders and in-out boxes. In designing an animation system for kids, Halgren, Fernandes, and Thomas (1995) found many kids to be unfamiliar with the metaphor of a frame-based film strip. It is helpful to choose metaphors that are familiar to kids, though kids often have success in learning interfaces based on unfamiliar metaphors if they are clear and consistent (Schneider 1996).

36.2.5 Interaction Style

Children's patterns of attention and interaction are quite different from those of adults. Traditional task-oriented analyses of activity may fail to capture the playful, spontaneous nature of children's interactions with technology. For example, when adults are the intended users, designers take great pains to create error messages that are informative and understandable based on the assumption that users want to *avoid* generating the message again. Hanna, Risden, and Alexander (1997) used a funny noise as an error message and found that the children repeatedly generated the error to hear the noise.

Halgren and colleagues (1995) found that children would click on any readily visible feature just to see what would happen, and they might click on it repeatedly if it generated sound or motion in feedback. This behavior was causing young users to get trapped in advanced modes they did not understand. The designers chose to hide advanced functionality in drawers—a metaphor that is familiar to children. "By hiding the advanced tools, the novice users would not stumble onto them and get lost in their functionality. Rather, only the advanced users who might want the advanced tools would go looking for more options. This redesign allows the product to be engaging and usable by a wider range of ages and abilities" (Halgren, Fernandes, and Thomas 1995). Resnick and Silverman go even further in their design principle: "make it as simple as possible—and maybe even simpler" (Resnick and Silverman 2005). They warn against "functionality creep" and suggest removing advanced functionality altogether if it is not clearly required to support children's creative efforts.

Children are more likely than adults to work with more than one person at a single computer. They enjoy doing so to play games (Inkpen 1997) and may be forced to do so because of limited resources in school (Stewart et al. 1998). Teachers may also create a shared-computer setup to promote collaborative learning. When multiple children work at one machine simultaneously, they need to negotiate sharing control of input devices. Giving students multiple input devices increases their productivity and their satisfaction (Inkpen 1997; Inkpen et al. 1995; Stewart et al. 1998). For use in developing countries where resources may be scarcer, Amershi et al. developed a text entry mechanism for multiple mice (Amershi et al. 2010).

Children also bring unique interaction styles to online environments; they respond to information they encounter while browsing the web in markedly different ways than adults. In a study of 55 first through fifth graders, the Nielsen Norman Group found that kids were often unable to distinguish between site content and advertisements. Moreover, they rarely scrolled down to find content; instead they chose to interact with site elements that were immediately visible. When examining a new site, children were willing to hunt for links in the content by "scrubbing the screen" with the mouse instead of relying solely on visual cues (Gilutz and Nielsen 2002). Using search on the web presents some unique challenges for children. Druin et al. found that children did not look at the screen while typing their search query, missing out on auto complete features that could help correct spelling or lead them to search results (Druin et al. 2009). Children also had difficulty constructing queries that required more than one search step.

36.3 CHILDREN WITH SPECIAL NEEDS

Recent research developments have begun to develop technology interactions for children with special needs. Disabled people represent 10% of the worlds' population (United Nations 2006) and in the United States, 14% of all children enrolled in public secondary education have special needs (National Center for Education Statistics 2011). The work in this area is mainly composed of technologies that foster children's development and education, and certainly there is a need for such focus. Some children with special needs are more likely to be behind in cognitive development than other children due to a lack of appropriate learning resources (Mayberry 2007). For example, more than 90% of deaf children are born to parents of hearing who do not know how to sign; yet language acquisition is crucial for early cognitive development (Moeller 2000). The philosophy that governs most designers in this area is one that focuses on designing for children's abilities rather than one that designs around their disabilities. There has also been a push to consider the abilities of children with special needs and examine how they can be included in the design process (Guha, Druin, and Fails 2008). In this section, we will discuss different technologies that have been developed for children with special needs, specifically for visual impairments, hearing and speech impairments, autistic children, motor and learning impairments, as well as the methods that have been employed for each.

36.3.1 Visually Impaired Children

Sánchez and Sáenz, who represent the majority of work in educational technologies for visually impaired and blind children, developed audio-based learning environments that focus on mathematics and memory (Sánchez and Sáenz 2005). Sánchez and Sáenz also built a mobile haptic and sound device for learning orientation and mobility skills (Sánchez, Sáenz, and Ripoll 2009). Looking to design for abilities, McElligott and van Leeuwan partnered with blind children and designed toys with both tactile and audio interactions, taking advantage of multiple stimuli (McElligott and van Leeuwen 2004). Similarly, Patomaki et al. used haptic feedback from an off-the-shelf game pad to develop a game for visually impaired children to assist in memory tasks (Patomäki et al. 2004). Catering toward more informal learning environments, Bruce and Walker assigned music and sounds to the dynamic position of fish in the Georgia Aquarium, allowing visually impaired visitors to be able to experience the aquarium in a different way (Walker et al. 2006).

36.3.2 Speech and Hearing Impairments

For children with speech impairments some technologies have assisted in speech development. Fell et al. built a system visiBabble which visualizes infant vocalizations in real time (Fell et al. 2004). The system reinforces the production of syllabic noises which has been associated with later language and cognitive development. Similarly, for children, Balter developed a speech training technology designed for children in speech therapy which helped detect the inaccuracies in mispronunciation. (Bälter et al. 2005).

For deaf children, some technologies have focused on learning sign language. Brashear and Henderson designed a game, CopyCat, which teaches deaf children to learn American Sign Language (Brashear et al. 2006; Henderson et al. 2005). The game uses gesture recognition technology to help young deaf children practice sign language skills. Focusing on comprehension of verbal language, Gennari and Mich developed an intelligent web system to teach temporal reasoning, a skill that is difficult for deaf children (Gennari and Mich 2008). Other novel interaction techniques that push the boundaries of technical innovation have been developed for deaf children. Iversen et al. developed a program called Stepstone, which uses body motion and group collaboration in order for children with cochlear implants to close the gap between linguistics and body movement (Iversen et al. 2007).

In designing for both speech and hearing impairments, the Wizard of Oz technique has been proven to be effective to evaluate these technologies. In this technique, children think that they are interacting with a computer, but instead of a computer, an adult interprets and provides feedback to the computer. This technique is helpful for gesture or voice recognition which may have a high error rate in early stages of development which may be frustrating for children who are testing the system. Both Balter and Brashear used this technique in the design of their systems (Bälter et al. 2005; Brashear et al. 2006).

36.3.3 Autistic Children

Autism is a neurodevelopment disorder that exhibits itself in abnormal social interaction, communication ability, and patterns of interests and behaviors. When presented with many choices, children with autism have difficulty in deciding which event is important and easily become overwhelmed. However, computers can also provide reliable and consistent rule enforcement which is useful for children who are uncomfortable with unpredictable environments.

There is a burgeoning area of work around using technology to help autistic children learn social skills. Piper et al. developed a cooperative game that runs on tabletop technology, which was found to be an effective tool for group work with teenagers on the autistic spectrum (Piper et al. 2006). De Leo and Leroy developed a mobile application to help autistic kids manage social interactions in situ (Leo and Leroy 2008). Merryman et al. use virtual peers for autistic children to learn social skills (Merryman et al. 2008). Putnam and Chong surveyed autistic children and parents and found that designing for the strengths of autistic children would be worthwhile; this includes abilities in math and reading (for children in the 7–10 age range), good memory, and a desire to be social (Putnam and Chong 2008).

36.3.4 Learning Impairments

In order to detect motor and learning disabilities early on, Westeyn developed different toys with multiple sensors to evaluate five levels of object play (Westeyn et al. 2008). These toys can help parents identify different motor and learning milestones by automatically collecting information about different kinds of play. To evaluate websites for children with learning disabilities, Andersen and Rowland added to a suite of existing accessibility tools to determine the cognitive load of a website (Andersen and Rowland 2007). Websites that have high reading levels, inconsistent navigation, pop-ups, are especially difficult for children with learning disabilities to navigate and understand. This tool helps developers to identify whether their design may cause problems for children with learning disabilities.

Specifically, for children with Down syndrome, Feng et al. surveyed 600 children with Down syndrome to explore technology use and potential design considerations for this population (Feng et al. 2010). They found that the children not only had cognitive difficulties such as reading and navigating, but also physical difficulties with controlling input devices, and security and privacy issues such as downloading viruses and releasing information. These results indicate that further work is needed in designing physical devices, information presentation, and addressing security and privacy issues for this population of users.

There is little work that addresses these needs for children with Down syndrome. One exception is Ortega-Tudela and Gomez-Ariza who built a system to teach basic mathematics using multimedia. In evaluating the system, they found that these tools aided learning better than paper and pencil (Ortega-Tudela and Gomez-Ariza 2006).

36.3.5 Motor Impairments

For children with motor impairments, technology offers opportunities to interact with their environment differently. EyeDraw, an eye-tracking application, enables children with severe motor impairments to create drawings using their eyes (Hornof and Cavender 2005). Similarly, VoiceDraw uses variations in vocal tones to generate free-form drawings (Harada, Wobbrock, and Landay 2007). Hornof describes a project in which he works with children that have severe motor impairments as design partners to learn how they wish to communicate (Hornof 2008). To address different input needs, Harada et al. developed a vocal joystick which allows users to control the mouse cursor using a variety of vocal parameters (Harada et al. 2006).

36.3.6 Hospitalized Children

In the sterile environment of hospitals, children are often isolated from their family and have few contacts with people outside of the hospital. At the same time, they undergo stressful and severe treatments. Technologies can provide opportunities to socialize and offer support. Tarrin et al. partnered with hospitalized children to create haptic games to be played with other people (Tarrin, Petit, and Chêne 2006). Others have created robotic companions for hospitalized children to mitigate fear and loneliness (Stiehl et al. 2009). Virtual environments also offer children opportunities to socialize within hospital settings. Bers describes an online environment where children undergoing dialysis are able to explore their identity and socialize online (Bers, Gonzalez-Heydrich, and DeMaso 2001).

Considering the abilities of special needs children not only improves accessibility to more children, but also helps push technological interaction innovation and methods.

36.4 CHILDREN AND THE DESIGN PROCESS

Users play a variety of roles in HCI design processes. Visionary designers such as Alan Kay and Seymour Papert began considering the abilities and sensibilities of children in the design of new technologies as early as the 1970s (Kay 1972; Papert 1972). Today, ethnographic and participatory (Schuler and Namioka 1993) methods are becoming increasingly common features of the human-centered design toolkit as HCI designers attempt to deeply understand the practices and preferences of people who will be using new technologies. When designers enter the world of children, and, conversely, when children enter the laboratory, many of the traditional rules change. As we have seen, children are not just “little adults”; they engage with the world in fundamentally different ways. Naturally, they bring a host of social, emotional, and cognitive elements to the design process that are unfamiliar to designers who are accustomed to working with adults. In this section, we examine new and traditional methods for working with children in a human-centered design process.

36.4.1 Use of Video with Children

Like adults, children may change their behavior when a video camera is present. Druin (1999) and her design team found in their early work that children tended to "freeze" or "perform" when they saw a video camera in the room (1999). In subsequent work, Druin's team observed that the problems associated with videotaping had more to do with power relationships than with the video cameras themselves. When the children are in control of the cameras, their discomfort decreases (Alborzi et al. 2000). In addition to considering the social impact of using a camera with children, there are also technical difficulties to deal with. Her research team found that, even with smaller cameras, it was difficult to capture data in small bedrooms and large public spaces. The sound and speech captured in public spaces was difficult to understand or even inaudible. Finally, it was difficult to know where to place cameras because they did not know where children would sit, stand or move in the environment. Druin recommends using multiple data sources to capture "messy" design environments with children, including notetakers and participant observers in addition to videotaping. She also encourages her design team to use video cameras (along with journal writing, team discussion, and adult debriefing) as a way to record their brainstorming sessions and other design activities.

Goldman-Segall (1996) explains why video data are an important part of ethnographic interviews and observations. When using video, the researcher does not have to worry about remembering or writing down every detail: "She can concentrate fully on the person and on the subtleties of the conversation." The researcher also has access to "a plethora of visual stimuli which can never be 'translated' into words in text," such as body language, gestures, and facial expressions. It is especially important to be able to review the body language of children as they interact with software. Hanna, Risden, and Alexander (1997) state that children's "behavioral signs are much more reliable than children's responses to questions about whether or not they like something, particularly for younger children. Children are eager to please adults, and may tell you they like your program just to make you happy." MacFarlane, Sim, and Horton (2005) suggest that both signs (behaviors) and symptoms (children's direct responses) should be used together to understand children's enjoyment of and ability to use new technologies. Video is extremely useful in being able to study behavioral signs as the researchers may miss some important signs and gestures during the actual observation or interview.

Instead of using video in its traditional capacity for ethnographic-style observation, some researchers have attempted to capitalize on children's playful treatment of video cameras to elicit articulation about new technologies. In studies using video probes to capture domestic communication patterns, Hutchinson et al. (2003) observe that images are particularly attractive to young people as an entertaining medium for interacting and communicating. During classroom observations of children using math-learning software, Lamberty and Kolodner (2005) encouraged children to engage in "camera talk" with stationary cameras if they wished. Many of the children regularly talked to the camera. This spontaneous behavior revealed both their preferences for using the software and their developing understanding of fractions. Likewise, Iversen (2002) suggests that, by provoking children to verbalize, video cameras provide a communication link between designers and young informants, thereby enriching both the data collected and the design experience.

36.4.2 Methods for Designing and Testing with Kids

In this section, we review a variety of methods for designing with and for children. These methods differ dramatically in the amount of power they grant to children. Some methods encourage us to view children as codesigners with an equal voice in determining design direction, whereas others place children in a more reactive role as evaluators or subjects in laboratory-based usability tests. In practice, designers use methods from different points on this power spectrum depending on the maturity of the project, and often move back and forth between testing with kids and open-ended exploration (Scaife et al. 1997).

Druin unpacks this spectrum of control by describing four different roles that children can play in the design of new technologies: user, tester, informant, and design partner (Druin 2002). The most reactive role she describes for children in design is *user.* As *users*, children interact with existing technologies and have no direct impact on the design of the technology, except in the form of recommendations for future designs. As *testers*, children are asked to provide feedback about technology in development so that it can be refined before it is released; however, the goals of the technology itself are determined much earlier by adult designers. As *informants*, children play an earlier, more active role in determining the goals and features of new technologies. When children play the role of *informant*, they interact directly with designers, but ultimately, the designer decides what the children need or want based on observations, interviews or other data collection methods. Finally, Druin explains that as *design partners*, children are seen as equal stakeholders in the design process. Although they may not be able to contribute to the development of the technology in equivalent ways, their expertise is viewed as equal in importance to that of other contributors to the design process.

The notion of children as design partners will be explored more fully in Section 36.4.2.1. Methods for including children as informants and design partners borrow from the tradition of participatory design that emerged in the Scandinavian workplace. Participatory design is an "approach toward computer systems design in which the people destined to *use* the system play a critical role in *designing* it" (Schuler and Namioka 1993). With children, this idea is even more important: since they are physically and cognitively different from adults, their participation in the design process may offer significant insights. Schuler writes:

> [Participatory Design] assumes that the workers themselves are in the best position to determine how to improve their work and their work life... It views the users' perceptions of technology as being at least as important to success as fact, and their feelings about technology as at least as important as what they can do with it. (Schuler and Namioka 1993, p. xi)

Empowering children in this way and including them in the design process can be difficult due to the traditionally unequal power relationships between kids and adults.

On the other end of the spectrum, methods for including children as users and testers often borrows from the traditional practices of experimental psychology. Usability testing generally takes place in a controlled setting. Sometimes a single design is tested with the goal of improving it; at other times, different design ideas might be compared to establish which ones generate more positive feedback or better enable task completion. Data collection methods like verbal protocol analysis (Ericsson and Simon 1993) are commonly used and will be further discussed in the Section 36.4.2.2.

36.4.2.1 Cooperative Inquiry

Druin (1999) has developed a systematic approach to developing new technologies for children with children; she has created new research methods that include children in various stages of the design process. This approach, called *cooperative inquiry*, is a combination of participatory design, contextual inquiry, and technology immersion. Children and adults work together on a team as research and design partners. She reiterates the idea that "Each team member has experiences and skills that are unique and important, no matter what the age or discipline" (Alborzi et al. 2000).

In this model, the research team frequently observes children interacting with software, prototypes, or other devices to gain insight into how child users will interact with and use these tools. When doing these observations, adult and child researchers both observe, take notes, and interact with the child users. During these observations, there are always at least two notetakers and one interactor, and these roles can be filled by either an adult or child team member. The interactor is the researcher who initiates discussion with the child user and asks questions concerning the activity. If there is no interactor or if the interactor takes notes, the child being observed may feel uncomfortable, like being "on stage" (Druin 1999). Other researchers have also found that the role of interactor can be useful for members of the design team. Scaife and Rogers (1999) successfully involved children as informants in the development of ECOi, a program that teaches children about ecology. They wanted the kids to help them codesign some animations in ECOi. Rather than just having the software designer observe the children as they played with and made comments about the ECOi prototypes, the software designer took on the role of interactor to elicit suggestions directly. Through these on-the-fly, high-tech prototyping sessions, they learned that "it was possible to get the software designer to work more closely with the kids and to take on board some of their more imaginative and kid-appealing ideas" (Scaife and Rogers 1999).

When working as design partners, children are included from the beginning. The adults do not develop all the initial ideas and then later see how the children react to them. The children participate from the start in brainstorming and developing the initial ideas. The adult team members need to learn to be flexible and learn to break away from carefully following their session plans, which is too much like school. Children can perform well in this more improvisational design setting, but the extent to which the child can participate as a design partner depends on his/her age. Children younger than 7 years may have difficulty in expressing themselves verbally and being self-reflective. These younger children also have difficulty in working with adults to develop new design ideas. Children older than 10 are typically beginning to become preoccupied with pre-conceived ideas of the way "things are supposed to be." In general, it has been found that children of age 7–10 years are the most effective prototyping partners. They are "verbal and self-reflective enough to discuss what they are thinking," and understand the abstract idea that their low-tech prototypes and designs are going to be turned into technology in the future. They also do not get bogged down with the notion that their designs must be similar to preexisting designs and products.

Through her work with children as design partners, Druin (1999) and Druin et al. (2001) has discovered that there are stumbling blocks on the way to integrating children into the design process and to helping adults and children work together as equals. One set of problems deals with the ability of children to express their ideas and thoughts. When the adult and children researchers are doing observations, it is best to allow each group to develop its own style of note-taking. Adults tend to take detailed notes, and children tend to prefer to draw cartoons with short, explanatory notes. It is often difficult to create one style of note-taking that will suit both groups. Since children may have a difficult time communicating their thoughts to adults, low-tech prototyping is an easy and concrete way for them to create and discuss their ideas. Art supplies such as paper, crayons, clay, and string allow adults and children to work on an equal footing. A problem that arises in practice is that since these tools are child-like, adults may believe that only the child needs to do such prototyping. It is important to encourage adults to participate in these low-tech prototyping sessions.

The second set of problems emerges from the traditionally unequal power relationships between adults and children. In what sense can children be treated as peers? When adults and children are discussing ideas, making decisions, or conducting research, traditional power structures may emerge. In conducting a usability study, the adult researcher might lead the child user through the experiment rather than allowing the child to explore freely on his/her own. In a team discussion, the children may act as if they are in a school setting by raising their hands to speak. Adults may even inadvertently take control of discussions. Is it sensible to set up design teams where children are given equal responsibilities to those of adult designers? Getting adults and children to work together as a team of equals is often the most difficult part of the design process. It is to be expected that it may take a while for a group to become comfortable and efficient when working together. It can take up to 6 months for an "intergenerational design team to truly develop the ability to build upon each other's ideas" (Druin et al. 2001). To help diffuse such traditional adult-child relationships, adults are encouraged to dress casually, and there always should be more than one adult and more than one child on a team. A single child may feel outnumbered by the adults, and a single adult might create the feeling of a school environment where the adult takes on the role of teacher. Alborzi et al. (2000) starts each design session with 15 minutes of snack time, where adults and children can informally discuss anything. This helps both adults and children to get to know each other better as "people with lives outside of the lab" (Alborzi et al. 2000) and to improve communication within the group.

Scaife et al. (1997) identified aspects of working with children in the role of informant that require special attention. They found that, when working in pairs, children feel less inhibited about telling strange adults what they were thinking. Other researchers have also found that pairing children, especially with friends, can help ease discomfort (Dindler et al. 2005; Als, Jensen, and Skov 2005). Scaife et al. (1997) caution that adults, too, need to become comfortable in the role of facilitator and should take care not to intervene too quickly if children's discussions wander.

In addition to the social challenges associated with mixing adults and children as equal design partners, kids do not always know how to collaborate well with one another in the first place. Because collaborating on a design project is often a novel experience for children, organizing the activities (without imposing too rigid a structure) can help create productive sessions. For example, Guha et al. (2005) describe a technique to support collaboration among kids and adults during cooperative inquiry sessions called "mixing ideas." First, kids generate ideas in a one-on-one session with an adult facilitator, then work in small groups to integrate these ideas, then in larger groups until the whole group is finally working together.

Although there have been many successes in having children participate as design and research partners in the development of software, there are still many questions to be answered about the effectiveness of this approach. Scaife and Rogers (1999) attempt to address many of the questions and problems faced when working with children in their work on informant design. The first question deals with the multitude of ideas and suggestions produced by children. Children say outrageous things. How do you decide which ideas are worthwhile? When do you stop listening? The problem of selection is difficult since in the end it is the adult who will decide which ideas to use and which ideas to ignore. Scaife and Rogers suggest creating a set of criteria to "determine what to accept and what not to accept with respect to the goals of the system… You need to ask what the trade-offs will be if an idea or set of ideas are implemented in terms of critical 'kid' learning factors: that is, how do fun and motivation interact with better understanding?" (Scaife and Rogers 1999)

In addition to deciding which of the children's ideas to use, there is also the problem of understanding the meaning behind what the child is trying to say. Adults tend to assume that they can understand what kids are getting at, but kid talk is not adult talk. It is important to remember that children have "a different conceptual framework and terminology than adults" (Scaife and Rogers 1999).

Another problem with involving children, particularly with the design of educational software, is that "children can't discuss learning goals that they have not yet reached themselves" (Scaife and Rogers 1999). Can children make effective contributions about the content and the way they should be taught, something which adults have always been responsible for? Adults have assumptions about what is an effective way to teach children. Kids tend to focus on the fun aspects of the software rather than the educational agenda. There may exist a mismatch of expectations if kids are using components of the software in unanticipated ways. Involving children in the design and evaluation process may help detect where these mismatches occur in the software.

36.4.2.2 Adapting Usability Evaluation Methods

HCI practices have evolved to address usefulness, enjoyability, and other measures of design success; however, usability remains a fundamental concern for HCI designers. Although efficiency and task completion are often not central to kids' goals in using technology, usability problems can create barriers to achieving other goals. For example, much research done to date has focused on designing educational software, and evaluation is primarily of learning outcomes, not usability. However, usability is a prerequisite for learning. In student projects in Georgia Tech's graduate class "Educational Technology: Design and Evaluation," many student designers never are able to show whether the educational design of their software is successful. What they find instead is that usability problems intervene, and they are unable to even begin to explore pedagogical efficacy. If children cannot use educational technology effectively, they certainly will not learn through the process of using it. MacFarlane, Sim, and Horton (2005) found that measurements of usability and "fun" were significantly correlated in studies of educational software for science. Usability is similarly important for entertainment, communications, and other applications. Many researchers have explored the effectiveness of traditional usability methods with children. However there is little work that adapts usability methods for tangible and mobile devices. We anticipate this area to grow with the increasing popularity of these devices. In this section, we examine comparative assessments of usability methods and review findings and recommendations.

36.4.2.2.1 Traditional Usability Testing

Several guidelines developed for work with adults become more important when applied to children. For example, when children are asked to work as testers, it is important to emphasize that it is the software that is being tested, not the participant (Rubin 1994). Children might become anxious at the thought of taking a test, and test taking may conjure up thoughts of school. The researcher can emphasize that even though the child is participating in a test, the child is the tester, not the one being tested (Hanna, Risden, and Alexander 1997). Rubin recommends that you show the participant where video cameras are located, let them know what is behind the one-way mirror, and whether people will be watching. With children, showing them behind the one-way mirrors and around the lab gives them "a better sense of control and trust in you" (Hanna, Risden, and Alexander 1997).

Markopoulos and Bekker (2002) describe the following characteristics of kids that can impact the process and outcome of usability testing:

- Children's capacity to verbalize thoughts is still developing.
- Personality may impact both kid's willingness to speak up to adults and their motivation to please authority figures.
- The capacity to concentrate is variable among kids.
- Young children are still developing the capacity for abstract and logical thinking; they may differ in cognitive ability such as remembering several items at once.
- The ability to monitor goal-directed performance develops throughout childhood and adolescence.
- Gender differences may be more pronounced at some ages than others.
- With small children, basic motor skills ability may be a barrier to effective evaluation if kids cannot use prototypes with standard input devices.

Hanna, Risden, and Alexander (1997) developed the following set of guidelines for laboratory-based usability testing with children:

- The lab should be made a little more child-friendly by adding some colorful posters, but avoid going overboard as too many extra decorations may become distracting to the child.
- Try to arrange furniture so that children are not directly facing the video camera and one-way mirror, as the children may choose to interact with the camera and mirror rather than doing the task at hand.
- Children should be scheduled for an hour of lab time. Preschoolers will generally only be able to work for 30 minutes but will need extra time to play and explore. Older children will become tired after an hour of concentrated computer use, so if the test will last longer than 45 minutes, children should be asked if they would like to take a short break at some point during the session.
- Hanna and colleagues suggest that you "explain confidentiality agreements by telling children that designs are 'top-secret'." Parents should also sign the agreements, since they will inevitably also see and hear about the designs.
- Children up to 7 or 8 years will need a tester in the room with them for reassurance and encouragement. They may become agitated from being alone or following directions from a loudspeaker. If a parent will be present in the room with the child, it is important to explain to the parent that he/she should interact with the child as little as possible during the test. Older siblings should stay in the observation area or a separate room during the test as they may eventually be unable to contain themselves and start to shout out directions.
- Hanna suggests that you should "not ask children if they want to play the game or do a task—that gives them the option to say no. Instead use phrases such as 'Now I need you to …' or 'Let's do this …' or 'It's time to …'."

36.4.2.2.2 Think/Talk Aloud

An important method for collecting usability data with adults is think-aloud protocols. Think aloud protocols in HCI research are related to verbal protocol analysis methods in psychology, in which subjects are asked to describe what they are thinking about and paying attention to while they complete some set of tasks (Ericsson and Simon 1993). In usability tests, think aloud methods are generally used in concert with direct observation (Nielsen 1993). Researchers who have used this method with children have observed that children may make very few comments during testing (Donker and Reitsma 2004). In some cases, they seem to have difficulty with concurrent verbalization—verbalizing thoughts while they complete tasks. The cognitive load associated with learning and executing task itself might interfere with kids' abilities to talk about it (Hoysniemi, Hamalainen, and Turkki 2003).

Despite potential obstacles, it has been demonstrated that verbal comments from children can play an important role in identifying usability problems. Donker and Reitsma (2004) reported that, although children produced fewer comments, those few comments provided important information about the severity of usability problems that were identified by direct observation. Other work likewise suggests that, although using think aloud with kids may result in fewer utterances than other approaches, it can be used to generate useful usability data with both older kids aged 8–14 (Donker and Markopoulos 2002; Baauw and Markopoulos 2004) and younger children aged 6–7 (van Kesteren et al. 2003).

Active intervention is closely related to think aloud protocols but involves investigators asking planned questions to encourage testers to reflect aloud on actions at specific points while completing a task. In a small comparative study with kids ages 6 and 7, van Kesteren et al. (2003) found that active intervention elicited the most comments when compared to think aloud, posttask/retrospection, codiscovery, peer tutoring, and traditional usability testing. However, the effectiveness of different usability methods depends on the context and environment. When comparing Active Intervention, Peer

Tutoring, and Cross Age Tutoring, Edwards and Benedyk found that peer tutoring was the most effective method within a classroom setting (Edwards and Benedyk 2007); because children were used to working in pairs in the classroom, peer tutoring was a more authentic experience.

36.4.2.2.3 Posttask Interviews

In posttask interviews, testers are asked to describe their experiences after they have finished using a new technology to complete a set of tasks. In some cases, video data are reviewed with the participant to evoke comments. This kind of retrospective verbal protocol emerged from the same tradition as think aloud (Ericsson and Simon 1993). Van Kesteren et al. (2003) raise the question of whether young kids' limited capacity to hold in memory several concepts at once and still-developing ability to engage in abstract thought limit their ability to accurately recall and recount past actions. They found that some kids age 6 and 7 were able to recall past actions and describe the ways in which their understanding changed. They note that keeping things interesting is important; children become bored with reviewing videos unless the tasks themselves are engaging to watch. In studies with kids aged 9–11, Baauw and Markopoulos (2004) determined that posttask interviews alone revealed fewer usability problems than think aloud; however, when combined with data from observations, which is standard practice, there was no significant difference between the problems that the two methods revealed.

36.4.2.2.4 Codiscovery

Codiscovery exploration is a usability method that is used to understand users' experiences and perceptions of new product designs, especially those that may be unfamiliar. In codiscovery sessions, two users who know one another work together to perform a set of tasks using the product. The goal of using two acquainted users is to encourage them to talk about the problems they encounter and their perceptions of the product in the natural course of collaborating on a task instead of relying on a single user's verbal performance for an experimenter (Kemp and van Gelderen 1996). First, the two users are asked to figure out what a product does and compare it to other products they know about. Next, they are asked to collaborate on a set of specific tasks using the product. Finally, a discussion period allows designers to ask about observed problems and behaviors; in addition, participants can ask questions about the design and the intended purpose of the product. Van Kesteren et al. (2003) found that using codiscovery with kids aged 6 and 7 can be difficult because they often attempt to complete tasks individually. Even when seated next to one another, two children may not interact at all, resulting in very few comments about the product being tested. When compared with traditional usability tests, think aloud, posttask interviews, peer tutoring and active intervention, codiscovery was found to elicit the fewest comments from kids (van Kesteren et al. 2003).

36.4.2.2.5 Peer Tutoring

Peer tutoring is a method for usability testing that was developed to capitalize on the ways that children interact with one another in natural, playful settings. When children play together, they regularly teach one another games and invent rules of play. Hoysniemi, Hamalainen and Turkki (2003, p. 209) explain that "one definition of the usability of a children's software application is that a child is able and willing to teach other children how to use it." Instead of relying on task completion in a lab, peer tutoring is an approach to usability testing that allows kids to engage in exploratory and playful interactions in a naturalistic setting. Peer tutoring involves first helping one or more kids to develop expertise using a piece of software and then asking them to teach other kids how to use it. By observing, recording, and analyzing interactions between tutors and tutees, it is possible to identify usability problems in software as the kids attempt to teach it to one another. Hoysniemi, Hamalainen, and Turkki (2003) point out that, although it can be useful, the peer tutoring approach requires time, training, and careful implementation to be effective.

36.5 GENRES OF TECHNOLOGY FOR KIDS

Technology for kids falls into two broad categories: education and entertainment. When game companies try to mix these genres, they may use the term "edutainment." New products for kids increasingly include specialized hardware as well as software.

36.5.1 Entertainment

Designers of games and other entertainment software rarely write about how they accomplish their job. Talks are presented each year at the Game Developer's Conference (http://www.gdconf.com), and some informal reflections are gathered as conference proceedings. Attending the conference is recommended for people who wish to learn more about current issues in game design. The magazine *Game Developer* is the leading publication with reflective articles on the game design process.

Game designers are usually gamers themselves and often end up simply designing games that they themselves would like to play. This simple design technique is easy and requires little if any background research with users. Because most game designers have traditionally been male, this approach allowed them to appeal quite effectively to a core gaming audience: young men and teenage boys. However, female designers are becoming more common on design teams and gaming companies are increasingly recognizing that people outside the typical gamer stereotype represent a large potential market for their products. Designing for teenagers is relatively easy. As we have seen, designing for very young children presents substantial challenges. The younger your target audience, the more they should be tightly connected to every stage of the design process.

Brenda Laurel pioneered the use of careful design methods for non-traditional game audiences in her work with the company Purple Moon in the mid-1990s. Laurel aimed to develop games that appeal to pre-teen girls both to tap this market segment and also to give girls an opportunity to become fluent with technology. Many people believe that use of computer game leads to skills that later give kids advantages at school and work. Through extensive interviews with girls in their target age range, Purple Moon was able to create successful characters and game designs. However, the process was so time consuming and expensive that the company failed to achieve profitability fast enough to please its investors. The company was closed in 1999, and its characters and games were sold to Mattel. Purple Moon perhaps did more research than was strictly necessary, particularly because their area was so new. The broader lesson is that the game industry typically does not budget for needs analysis and iterative design early in the design process. "Play testing" and "quality assurance" typically take place relatively late in the design cycle. Designers contemplating incorporating research early in their design process must consider the financial cost. (For more on game design and evaluation, see Chapter 34.)

Oosterholt, Kusano, and Vries (1996) describe several design constraints that are specific to the design of products for children. First, they suggest that trying to be fashionable can result in products that are quickly perceived by kids as outdated. They also point out that "fun" is just as important to measure as usability, that measurements of fun should be shared with development teams and, moreover, that the product should grow with users over time and continue to be fun long after kids have learned to use it.

Game designer Carolyn Miller highlights the following seven mistakes ("kisses of death") commonly made by people trying to design games for kids:

> *"Death kiss #1: Kids love anything sweet"*
> Miller writes that "sweetness is an adult concept of what kids should enjoy." Only very young children will tolerate it. Humor and good character development are important ingredients. Don't be afraid to use off-color humor, or to make something scary.
>
> *"Death kiss #2: Give 'em what's good for 'em"*
> She advises, "Don't preach, don't lecture, and don't talk down—nothing turns kids off faster."
>
> *"Death kiss #3: You just gotta amuse 'em"*
> "Don't assume that just because they are little, they aren't able to consume serious themes."
>
> *"Death kiss #4: Always play it safe!"*
> Adult games often rely on violence to maintain dramatic tension. Since you probably won't want to include this in your game for kids, you'll need to find other ways to maintain dramatic tension. Don't let your game become bland.
>
> *"Death kiss #5: All kids are created equal"*
> Target a specific age group, and take into consideration humor, vocabulary, skill level, and interests. If you try to design for everyone, your game may appeal to no one.
>
> *"Death kiss #6: Explain everything"*
> In an eagerness to be clear, some people over-explain things to kids. Kids are good at figuring things out. Use as few words as possible, and make sure to use spoken and visual communication as much as possible.
>
> *"Death kiss #7: Be sure your characters are wholesome!"*
> Miller warns that if every character is wholesome, the results are predictable and boring. Characters need flaws to have depth. Miller identifies a number of common pitfalls in assembling groups of characters. It's not a good idea to take a "white bread" approach, in which everyone is white and middle class. On the other end of the spectrum, it's also undesirable to take a "lifesaver approach" with one character for each ethnicity. Finally, you also need to avoid an "off-the-shelf" approach, in which each character represents a stereotype: "You've got your beefy kid with bad teeth; he's the bully. You've got the little kids with glasses; he's the smart one." Create original characters that have depth and have flaws that they can struggle to overcome (Miller, 1998).

36.5.2 Education

To design educational software, we must expand the concept of user-centered design (UCD) to one of learner-centered design (LCD) (Soloway, Guzdial, and Hay 1994). There are several added steps in the process, as follows:

- Needs analysis
    - For learners
    - For teachers
- Select pedagogy
- Select media/technology
- Prototype
    - Core application
    - Supporting curricula
    - Assessment strategies
- Formative evaluation
    - Usability
    - Learning outcomes
- Iterative design
- Summative evaluation
    - Usability
    - Learning outcomes

In our initial needs analysis, for software to be used in a school setting, we need to understand not just learners but also teachers. Teachers have heavy demands on their time and are held accountable for their performance in ways that vary between districts and between election years.

Once we understand our learner and teacher needs, we need to select an appropriate *pedagogy*—an approach to teaching and learning. For example, behaviorism views learning as a process of stimulus and reinforcement (Skinner 1968). Constructivism sees learning as a process of active construction of knowledge through experience. A social-constructivist perspective emphasizes learning as a social process (Newman, Griffin, and Cole 1989). (A full review of approaches to pedagogy is beyond the scope of this chapter.)

Next, we are ready to select the media we will be working with, matching their affordances to our learning objectives and pedagogical approach. Once the prototyping process has begun, we need to develop not just software or hardware, but (for applications to be used in schools) also supporting curricular materials and assessment strategies.

“Assessment” should not be confused with “evaluation.” The goal of assessment is to judge an individual student’s performance. The goal of evaluation is to understand to what extent our learning technology design is successful. An approach to assessing student achievement is an essential component of any school-based learning technology. For both school and free-time use, we need to design feedback mechanisms so that learners can be aware of their own progress. It is also important to note whether learners find the environment motivating. Does it appeal to all learners, or more to specific gender, learning style, or interest groups?

As in any HCI research, educational technology designers use formative evaluation to informally understand what needs improvement in their learning environment, and guide the process of iterative design. Formative evaluation must pay attention first to usability, and second to learning outcomes. If students cannot use the learning hardware or software, they certainly will not learn through its use. Once it is clear that usability has met a minimum threshold, designers then need to evaluate whether learning outcomes are being met. After formative evaluation and iterative design are complete, a final summative evaluation serves to document the effectiveness of the design and justify its use by learners and teachers. Summative evaluation must similarly pay attention to both usability and learning outcomes.

A variety of quantitative and qualitative techniques are commonly used for evaluation of learning outcomes (Gay and Airasian 2000). Most researchers use a complementary set of both quantitative and qualitative approaches. Demonstrating educational value is challenging, and research methods are an ongoing subject of research.

This represents an idealized LCD process. Just as many software design projects do not in reality follow a comprehensive UCD process, many educational technology projects do not follow a full learner-centered design process. Learner-centered design is generally substantially more time consuming than UCD. While it may in some cases be possible to collect valid usability data in a single session, learning typically takes place over longer periods of time. To get meaningful data, most classroom trials take place over weeks or months. Furthermore, classroom research needs to fit into the school year at the proper time. If you are using Biologica (Hickey et al. 2000) to teach about genetics, you need to wait until it is time to cover genetics that school year. You may have only one or two chances per year to test your educational technology. It frequently takes many years to complete the learner-centered design process. In the research community, one team may study and evolve one piece of educational technology over many years. In a commercial setting, educational products need to get to market rapidly, and this formal design process is rarely used.

36.5.3 Genres of Educational Technology

In 1980, Taylor divided educational technology into three genres:

Computer as tutor
Computer as tool
Computer as tutee

Suppose that we are learning about acid rain. If the computer is serving as *tutor*, it might present information about acid rain and ask the child questions to verify the material was understood. If the computer is a *tool*, the child might collect data about local acid rain and input that data into an ecological model to analyze its significance. If the computer is a *tutee*, the child might program his or her own ecological model of acid rain.

With the advent of the Internet, we must add a fourth genre:

Computer-supported collaborative learning (CSCL)

In a CSCL study of acid rain, kids from around the country might collect local acid rain data, enter it into a shared database, analyze the aggregate data, and talk online with adult scientists who study acid rain. This is in fact the case in the Acid Rain Project (Tinker 1993). See Table 36.1 for an overview of genres of children’s software.

36.5.3.1 Computer as Tutor

In most off-the-shelf educational products, the computer acts as tutor. Children are presented with information and then quizzed on their knowledge. This approach to education is grounded in behaviorism (Skinner 1968). It is often referred to as “drill and practice” or “computer-aided instruction” (CAI). The computer tracks student progress and repeats exercises as necessary.

Researchers with a background in artificial intelligence have extended the drill and practice approach to create “intelligent tutoring systems.” Such systems try to model what the user knows and tailor the problems presented to an individual’s needs. Many systems explicitly look for typical mistakes and provide specially-prepared corrective feedback. For example, suppose a child adds 17 and 18 and gets an answer of 25 instead of 35. The system might infer that the child needs help learning to carry from the ones to the tens column and present a lesson on that topic. One challenge in the design of intelligent tutors is in accurately modeling what the student knows and what their errors might mean.

Byrne et al. (1999) has experimented with using eye tracking to improve the performance of intelligent tutors. Using an eye tracker, the system can tell whether the student has paid attention to all elements necessary to solve the problem. In early trials with the eye tracker, he found that some of the helpful hints the system was providing to the user were never actually read by most students. This helped

TABLE 36.1
Genres of Children's Software

Genre	Description
Entertainment	Games created solely for fun and pleasure.
Educational	Software created to help children learn about a topic using some type of pedagogy—an approach to teaching and learning.
Computer as tutor	Often referred to as "drill and practice" or "computer-aided instruction" (CAI), this approach is grounded in behaviorism. Children are presented with information and then quizzed on their knowledge.
Computer as tool	The learner directs the learning process, rather than being directed by the computer. This approach is grounded in constructivism, which sees learning as an active process of constructing knowledge through experience.
Computer as tutee	Typically, the learner uses construction kits to help reflect upon what he or she learned through the process of creation. This approach is grounded in constructivism and constructionism.
Computer-supported collaborative learning (CSCL)	Children use the Internet to learn from and communicate with knowledgeable members of the adult community. Children can also become involved in educational online communities with children from different geographical regions. This approach is grounded in social constructivism.
Edutainment	A mix of the entertainment and educational genres.

guide their design process. They were previously focusing on how to improve the quality of hints provided; however, that is irrelevant if the hints are not even being read (Byrne et al. 1999).

An interesting variation on the traditional "computer as tutor" paradigm for very young children is the Actimates line interactive plush toys. Actimates Barney and other characters lead children in simple games with educational value, like counting exercises. The "tutor" is animated and anthropomorphized. The embodied form lets young children use the skills they have in interacting with people to learn to interact with the system, enhancing both motivation and ease of use (Strommen 1998; Strommen and Alexander 1999).

36.5.3.1.1 Computer as Tool

When the computer is used as a tool, agency shifts from the computer to the learner. The learner is directing the process, rather than being directed. This approach is preferred by constructivist pedagogy, which sees learning as an active process of constructing knowledge through experience. The popular drawing program Kid Pix is an excellent example of a tool customized for kids' interests and needs. Winograd comments that Kid Pix's designer Craig Hickman "made a fundamental shift when he recognized that the essential functionality of the program lay not in the drawings that it produced, but in the experience for the children as they used it" (Winograd 1996). For example, Kid Pix provides several different ways to erase the screen—including having your drawing explode, or being sucked down a drain.

Simulation programs let learners try out different possibilities that would be difficult or impossible in real life. For example, Biologica (an early version was called "Genscope") allows students to learn about genetics by experimenting with breeding cartoon dragons with different inherited characteristics like whether they breathe fire or have horns (Hickey et al. 2000). Model-It lets students try out different hypotheses about water pollution and other environmental factors in a simulated ecosystem (Soloway et al. 1996).

The goal of such programs is to engage students in scientific thinking. The challenge in their design is how to get students to think systematically, and not simply try out options at random. Programs like Model-It provide the student with "scaffolding." Initially, students are given lots of support and guidance. As their knowledge evolves, the scaffolding is "faded," allowing the learner to work more independently (Guzdial 1994; Soloway, Guzdial, and Hay 1994).

36.5.3.1.2 Computer as Tutee

Seymour Papert comments that much CAI is "using the computer to program the child" (Papert 1992, p. 163). Instead, he argues that the child should learn to program the computer and through this process gain access to new ways of thinking and understanding the world. Early research argued that programming would improve children's general cognitive skills, but empirical trials produced mixed results (Clements 1986; Clements and Gullo 1984; Pea 1984). Some researchers argue that the methods of these studies are fundamentally flawed, because the complexity of human experience cannot be reduced to pre and post tests (Papert 1987). The counter-argument is that researchers arguing that technology has a transformative power need to back up their claims with evidence of some form, whether quantitative or qualitative

FIGURE 36.2 Penguins created using Hypergami.

(Pea 1987; Walker 1987). More recently, the debate has shifted to the topic of technological fluency. As technology increasingly surrounds our everyday lives, the ability to use it effectively as a tool becomes important for children's success in school and later in the workplace (Resnick and Rusk 1996).

In the late 1960s, Feurzeig and colleagues (Feurzeig, personal communication, 1996) at BBN invented Logo, the first programming language for kids. Papert extended Logo to include "turtle graphics," in which kids learn geometric concepts by moving a "turtle" around the screen (Papert 1980). A variety of programming languages for kids have been developed over subsequent years, including Starlogo (Resnick 1994), Boxer (diSessa and Abelson 1986), Stagecast (Cypher and Smith 1995), Agentsheets (Repenning and Fahlen 1993), MOOSE (Bruckman 1997), Squeak (Guzdial and Rose 2001), Alice (Cooper, Dann, and Pausch 2000), and Scratch (Resnick et al. 2009).

Additionally, several programming environments bridge the gap between physical constructions and representations. Lego Mindstorms (originally "Lego/Logo") is a programmable construction kit with physical as well as software components (Martin and Resnick 1993). PicoCrickets, a craft-based robotic construction kit, offers a different pathway into robotics by leveraging interests in craft (Rusk et al. 2008). Similarly, LilyPad Arduino allows microcontrollers and sensors to be sewn into fabric to create e-textiles (Buechley et al. 2008).

Finally, Hypergami, a computer-aided design tool for origami allows students to learn about both geometry and art (Figure 36.2) (Eisenberg, Nishioka, and Schreiner 1997).

In most design tools, the goal is to facilitate the creation of a product. In educational construction kits, the goal instead is what is learned through the process of creation. So what makes a good construction kit? In a 1996 *Interactions* article titled "Pianos, Not Stereos: Creating Computational Construction Kits," Resnick, Bruckman, and Martin discuss the art of designing construction kits for learning ("constructional design"):

> The concept of learning-by-doing has been around for a long time. But the literature on the subject tends to describe specific activities and gives little attention to the general principles governing what kinds of "doing" are most conducive to learning. From our experiences, we have developed two general principles to guide the design of new construction kits and activities. These constructional-design principles involve two different types of "connections":
>
> - *Personal connections.* Construction kits and activities should connect to users' interests, passions, and experiences. The point is not simply to make the activities more "motivating" (though that, of course, is important). When activities involve objects and actions that are familiar, users can leverage their previous knowledge, connecting new ideas to their pre-existing intuitions.
> - *Epistemological connections.* Construction kits and activities should connect to important domains of knowledge—more significantly, encourage new ways of thinking (and even new ways of thinking about thinking). A well-designed construction kit makes certain ideas and ways of thinking particularly salient, so that users are likely to connect with those ideas in a very natural way, in the process of designing and creating (Resnick, Bruckman, and Martin 1996).
>
> Bruckman adds a third design principle:
>
> - *Situated support.* Support for learning should be from a source (either human or computational) with whom the learner has a positive personal relationship, ubiquitously available, richly connected to other sources of support, and richly connected to everyday activities (Bruckman 2000).

Resnick and Silverman (2005) have suggested several more design principles for creating construction kits for kids. Some, such as "iterate, iterate—then iterate again," are familiar mantras for HCI designers. Others may be less familiar, as follows:

- Low Floor and Wide Walls
  If a technology has a low floor, it means that it is easy for novices to begin using it. Wide walls suggest a wide range of possible areas of design and exploration. Construction kits define "a place to explore, not a collection of specific activities."
- Make Powerful Ideas Salient—Not Forced
  When designing toward specific learning goals, construction kits should make these ideas visible and useful in design activities rather than imposing the ideas on students as a pre-determined solution.
- Support Many Paths, Many Styles
  Kids approach problems in different ways; it is important to support a variety of design approaches in a construction kit.
- Make It as Simple as Possible—and Maybe Even Simpler
  Constraints can be the designer's best friend. Limited functionality sometimes wins out over more sophisticated designs because simplicity allows kids to find creative new ways to use a product.

- Choose Black Boxes Carefully
  This principle is related to the previous one; deciding when to reveal complexity and when to conceal it is a difficult question. Resnick and Silverman again suggest that the simplest choice is often the best one.
- A Little Bit of Programming Goes a Long Way
  Because programming is the fundamental mode of construction with computers, designers of construction kits for kids often include some programming functionality. Focusing on powerful, simple commands that kids can do well is often the best way to support a diverse range of activities.
- Give People What They Want—Not What They Ask For
  Observations of kids can often tell designers more than their direct answers to questions. Kids may ask for unrealistic features or may not know themselves why they are having difficulty completing a task.
- Invent Things That You Would Want to Use Yourself
  Although they caution against overgeneralizing one's own personal likes and dislikes, Resnick and Silverman propose that the most respectful approach to designing for kids is to create something that the designer herself finds enjoyable.

36.5.3.2 Computer-Supported Collaborative Learning

Most tools for learning have traditionally been designed for one child working at the computer alone. However, learning is generally recognized to be a social process (Newman, Griffin, and Cole 1989). With the advent of the Internet come new opportunities for children to learn from one another and from knowledgeable members of the adult community. This field is called "CSCL" (Koschmann 1996).

CSCL research can be divided into four categories:

1. Distance education
   Attempts to use online environments in ways that emulate a traditional classroom.
2. Information retrieval
   Research projects in which students use the Internet to find information.
3. Information sharing
4. Technological samba schools

Students debate issues with one another. One of the first such tools was the Computer-Supported Intentional Learning Environment, a networked discussion tool designed to help students engage in thoughtful debate as a community of scientists does (Scardamalia and Bereiter 1994). They may also collect scientific data and share it with others online. In the "One Sky, Many Voices" project, students learn about extreme weather phenomena by sharing meteorological data they collect with other kids from around the world, and also by talking online with adult meteorologists (Songer 1996). In the Palaver Tree Online project, kids learn about history by talking online with older adults who lived through that period of history (Ellis and Bruckman 2001).

A key challenge in the design of information sharing environments is how to promote serious reflection on the part of students (Guzdial 1994; Kolodner and Guzdial 1996).

In *Mindstorms*, Seymour Papert has a vision of a "technological samba school." At samba schools in Brazil, a community of people of all ages gather together to prepare a presentation for carnival. "Members of the school range in age from children to grandparents and in ability from novice to professional. But they dance together and as they dance everyone is learning and teaching as well as dancing. Even the stars are there to learn their difficult parts" (Papert 1980). People go to samba schools not just to work on their presentations, but also to socialize and be with one another. Learning is spontaneous, self-motivated, and richly connected to popular culture. Papert imagines a kind of technological samba school where people of all ages gather together to work on creative projects using computers. The Computer Clubhouse is an example of such a school in a face-to-face setting (Resnick and Rusk 1996). MOOSE Crossing is an Internet-based example (Bruckman 1998). A key challenge in the design of such environments is how to grapple with the problem of uneven achievement among participants. When kids are allowed to work or not work in a self-motivated fashion, typically some excel while others do little. (Elliott et al. 2000).

36.5.4 Safety Issues

One challenge in the design of Internet-based environments for kids is the question of safety. The Internet does contain inappropriate content such as information that is sexually explicit, violent, and racist. Typically, such information does not appear unless one is looking for it; however, it is unusual but possible to stumble across it accidentally. Filtering software blocks access to useful information as well as harmful (Schneider 1997). Furthermore, companies that make filtering software often fail to adequately describe how they determine what to block, and they may have unacknowledged political agendas that not all parents will agree with. Resolving this issue requires a delicate balance of the rights of parents, teachers, school districts, and children (Electronic Privacy Information Center 2001).

36.5.4.1 Cyberbullying

Cyberbullying, or the use of information communication technologies to bully others, has been on the rise with the advent of social networking sites, blogs, mobile phones, online games, and other communication technologies (Lenhart 2007). Unlike physical bullying, bullies can be anonymous and harmful information or comments can be persistent, searchable, and have the potential to be read by a large audience. Cyberbullying tends to be the most common in middle and high school ages (Lenhart 2007). In a survey study of 177 seventh grade students, over a quarter of students reported that they had been cyberbullied and one out of six students had cyberbullied others (Li 2006). The victims were also 60% female, while a slim majority of males are cyberbullies; this study, along with others, point that females prefer to use digital mediums to bully others.

Research on cyberbullying victims show that there are indeed serious consequences. Victims have lower self-esteem, an increased risk of depression and suicide, and may become cyberbullies themselves (Li 2007).

36.5.4.2 Predators

Another danger for kids online is the presence of sexual predators and others who wish to harm children. While such incidents are rare, it is important to teach kids not to give out personal information online such as their last name, address, or phone number. Kids who wish to meet an online friend face to face should do so by each bringing a parent and meeting in a well-populated public place like a fast-food restaurant. A useful practical guide "Child Safety on the Information Superhighway" is available from the Center for Missing and Exploited Children (http://www.missingkids.com). Educating kids, parents, and teachers about online safety issues is an important part of the design of any online software for kids.

36.6 CONCLUSION

To design for kids, we must have a model of what kids are and what we would like them to become. Adults were once kids. Many are parents. Some are teachers. We tend to think that we know kids—who they are, what they are interested in, what they like. However, we do not have as much access to our former selves as many would like to believe. Furthermore, it is worth noting that our fundamental notions of childhood are in fact culturally constructed and change over time. Karin Calvert writes about the changing notion of childhood in America, and the impact it has had on artifacts designed for children and child-rearing:

> In the two centuries following European settlement, the common perception in America of children changed profoundly, having first held to an exaggerated fear of their inborn deficiencies, then expecting considerable self-sufficiency, and then, after 1830, endowing young people with an almost celestial goodness. In each era, children's artifacts mediated between social expectations concerning the nature of childhood and the realities of child-rearing: before 1730, they pushed children rapidly beyond the perceived perils of infancy, and by the nineteenth century they protected and prolonged the perceived joys and innocence of childhood. (Calvert 1992, p. 8)

While Calvert was reflecting on the design of swaddling clothes and walking stools, the same role is played by new technologies for kids like programmable Legos and drill and practice arithmetic programs: these artifacts mediate between our social expectations of children and the reality of their lives. If you believe that children are unruly and benefit from strong discipline, then you are likely to design CAI. If you believe that children are creative and should not be stifled by adult discipline, then you might design an open-ended construction kit like Logo or Squeak. In designing for kids, it is crucial to become aware of one's own assumptions about the nature of childhood. Designers should be able to articulate their assumptions, and be ready to revise them based on empirical evidence.

REFERENCES

Alborzi, H., A. Druin, J. Montemayor, M. Platner, J. Porteous, L. Sherman, A. Boltman et al. 2000. Designing StoryRooms: Interactive storytelling spaces for children. In *Proceedings of the Symposium on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 95–104. Brooklyn, NY.

AlphaBaby. 2009. About AlphaBaby. http://alphababy.sourceforge.net/ (accessed May 18, 2010).

Als, B. S., J. J. Jensen, and M. B. Skov. 2005. Comparison of think-aloud and constructive interaction in usability testing with children. In *Proceedings of the 2005 Conference on Interaction Design and Children*. Boulder, CO.

Amershi, S., M. R. Morris, N. Moraveji, R. Balakrishnan, and K. Toyama. 2010. Multiple mouse text entry for single-display groupware. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, 169–78. Savannah, GA: ACM.

Andersen, A., and C. Rowland. 2007. Improving the outcomes of students with cognitive and learning disabilities: Phase I development for a web accessibility tool. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, 221–2. Tempe, AZ: ACM.

Antle, A. N., M. Droumeva, and D. Ha. 2009. Hands on what?: Comparing children's mouse-based and tangible-based interaction. In *Proceedings of the 8th International Conference on Interaction Design and Children*, 80–8. Como, Italy: ACM.

Baauw, E., and P. Markopoulos. 2004. A comparison of think-aloud and post-task interview for usability testing with children. In *Proceedings of the 2004 Conference on Interaction Design and Children*, 115–6. College Park, MD.

Bälter, O., O. Engwall, A. M. Öster, and H. Kjellström. 2005. Wizard-of-Oz test of ARTUR: A computer-based speech training system with articulation correction. In *Proceedings of the 7th International ACM SIGACCESS Conference on Computers and Accessibility*, 43. Baltimore, MD: ACM.

BBC. 1997. *Teletubbies Press Release*.

Bederson, B., J. Hollan, A. Druin, J. Stewart, D. Rogers, and D. Proft. 1996. Local tools: An alternative to tool palettes. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 169–70. Seattle, WA.

Berkovitz, J. 1994. Graphical interfaces for young children in a software-based mathematics curriculum. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: Celebrating Interdependence*, 247–8. Boston, MA.

Bernard, M., M. Mills, T. Frank, and J. McKnown. 2001. *Which Fonts Do Children Prefer to Read Online?* (Winter 2001), [Web Newsletter]. Software Usability Research Laboratory (SURL).

Bers, M. U., J. Gonzalez-Heydrich, and D. R. DeMaso. 2001. Identity construction environments: Supporting a virtual therapeutic community of pediatric patients undergoing dialysis. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 380–7. Seattle, WA: ACM.

Brashear, H., V. Henderson, K. Park, H. Hamilton, S. Lee, and T. Starner. 2006. American sign language recognition in game development for deaf children. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*, 79–86. Portland, OR: ACM.

Bruckman, A. 1997. *MOOSE Crossing: Construction, Community, and Learning in a Networked Virtual World for Kids*. Unpublished PhD, MIT.

Bruckman, A. 1998. Community Support for constructionist learning. *Comput Support Coop Work* 7:47–86.

Bruckman, A. 2000. Situated support for learning: Storm's weekend with rachael. *J Learn Sci* 9(3):329–72.

Buechley, L., M. Eisenberg, J. Catchen, and A. Crockett. 2008. The LilyPad Arduino: Using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 423–32. Florence, Italy.

Byrne, M. D., J. R. Anderson, S. Douglass, and M. Matessa. 1999. Eye tracking the visual search of click-down menus. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: The CHI is the Limit*, 402–9. Pittsburgh, PA.

Calvert, K. 1992. *Children in the House: The Material Culture of Early Childhood, 1600–1900*. Boston: Northeastern University Press.

Clements, D. H. 1986. Effects of logo and CAI environments on cognition and creativity. *J Educ Psychol* 78(4):309–18.

Clements, D. H., and D. F. Gullo. 1984. Effects of computer programming on young children's cognition. *J Educ Psychol* 76(6):1051–8.

Cooper, S., W. Dann, and R. Pausch. 2000. Alice: A 3-D tool for introductory programming concepts. *J Comput Sci Coll* 15(5):107–16.

Cypher, A., and D. C. Smith. 1995. End user programming of simulations. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 27–34. Denver, CO.

Davenport, A., and A. Wood. 1997. *TeleTubbies FAQ*, [Website]. BBC Education. http://www.bbc.co.uk/cbeebies/teletubbies/grownups/faq.shtml.

Dindler, C., E. Eriksson, O. S. Iversen, M. Ludvigsel, and A. Lykke-Olesen. 2005. Mission from Mars: A method for exploring user requirements for children in a narrative space. In *Proceedings of the 2005 Conference on Interaction Design and Children*. Boulder, CO.

diSessa, A. A., and H. Abelson. 1986. Boxer: A reconstructible computational medium. *Commun ACM* 29(9):859–68.

Donker, A., and P. Markopoulos. 2002. A comparison of think-aloud, questionnaires and interviews for testing usability with children. In *Proceedings of HCI 2002*, 305–16.

Donker, A., and P. Reitsma. 2004. Usability testing with young children. In *Proceedings of the Conference on Interaction Design and Children*, 43–8. College Park, MD.

Dourish, P. 2001. *Where the Action Is: The Foundations of Embodied Interaction*. Cambridge, MA: MIT Press.

Druin, A. 1999. Cooperative inquiry: Developing new technologies for children with children. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: The CHI is the Limit*, 592–9. Pittsburgh, PA.

Druin, A. 2002. The role of children in the design of new technology. *Behav Inf Technol* 21(1):1–25.

Druin, A., B. Bederson, J. P. Hourcade, L. Sherman, G. Revelle, M. Platner, and S. Weng. 2001. Designing a digital library for young children: An intergenerational partnership. In *Proceedings of the Joint Conference on Digital Libraries*. Roanoke, VA.

Druin, A., E. Foss, L. Hatley, E. Golub, M. L. Guha, J. Fails, and H. Hutchinson. 2009. How children search the internet with keyword interfaces. In *Proceedings of the 8th International Conference on Interaction Design and Children*, 89–96. Como, Italy: ACM.

Edwards, H., and R. Benedyk. 2007. A comparison of usability evaluation methods for child participants in a school setting. In *Proceedings of the 6th International Conference on Interaction Design and Children*, 9–16. Aalborg, Denmark: ACM.

Eisenberg, M., A. Nishioka, and M. E. Schreiner. 1997. Helping users think in three dimensions: Steps toward incorporating spatial cognition in user modeling. *Proceedings of the International Conference on Intelligent User Interfaces*, 113–20. Orlando, FL.

Electronic Privacy Information Center. 2001. *Filters and Freedom 2.0: Free Speech Perspectives on Internet Content Control*. Washington, DC: Electronic Privacy Information Center.

Elliott, J., A. Bruckman, E. Edwards, and C. Jensen. 2000. Uneven achievement in a constructionist learning environment. In *Proceedings of the International Conference on the Learning Sciences*, 157–63. Ann Arbor, MI.

Ellis, J. B., and A. S. Bruckman. 2001. Designing palaver tree online: Supporting social roles in a community of oral history. In *Proceedings of CHI: Conference on Human Factors in Computing Systems*, 474–81. Seattle, WA.

Erickson, T. 1990. Working with interface metaphors. In *The Art of Human-Computer Interface Design*, ed. B. Laurel, 65–73. Reading, MA: Addison Wesley Publishing Company Inc.

Ericsson, K. A., and H. Simon. 1993. *Protocol Analysis: Verbal Reports as Data*. Cambridge, MA: MIT Press.

Fell, H., C. Cress, J. MacAuslan, and L. Ferrier. 2004. visiBabble for reinforcement of early vocalization. In *Proceedings of the 6th International ACM SIGACCESS Conference on Computers and Accessibility*, 161–8. Atlanta, GA: ACM.

Feng, J., J. Lazar, L. Kumin, and A. Ozok. 2010. Computer usage by children with down syndrome: Challenges and future research. *ACM Trans Access Comput* 2(3):1–44.

Gay, L. R., and P. Airasian. 2000. *Education Research: Competencies for Analysis and Application*, 6th ed. Upper Saddle River, NJ: Merrill.

Gennari, R., and O. Mich. 2008. Designing and assessing an intelligent e-tool for deaf children. In *Proceedings of the 13th International Conference on Intelligent User Interfaces*, 325–8. Gran Canaria, Spain: ACM.

Gilutz, S., and J. Nielsen. 2002. Usability of websites for children: 70 design guidelines based on usability studies with kids. Nielsen Norman Group Report.

Goldman-Segall, R. 1996. Looking through layers: Reflecting upon digital video ethnography. *JCT Interdiscip J Curriculum Stud* 13(1).

Guha, M. L., A. Druin, G. Chipman, J. A. Fails, S. Simms, and A. Farber. 2005. Working with young children as technology design partners. *Commun ACM* 48(1):39–42.

Guha, M. L., A. Druin, and J. A. Fails. 2008. Designing with and for children with special needs: An inclusionary model. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 61–4. Chicago, IL: ACM.

Guzdial, M. 1994. Software-realized scaffolding to facilitate programming for science learning. *Interact Learn Environ* 4(1):1–44.

Guzdial, M., and K. Rose, ed. 2001. *Squeak: Open Personal Computing and Multimedia*: Prentice Hall.

Halgren, S., T. Fernandes, and D. Thomas. 1995. Amazing Animation™: Movie making for kids design briefing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 519–25. Denver, CO.

Hanna, L., K. Risden, and K. Alexander. 1997. Guidelines for usability testing with children. *Interactions* 4(5):9–14.

Harada, S., J. A. Landay, J. Malkin, X. Li, and J. A. Bilmes. 2006. The vocal joystick: Evaluation of voice-based cursor control techniques. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*, 197–204. Portland, OR: ACM.

Harada, S., J. O. Wobbrock, and J. A. Landay. 2007. Voicedraw: A hands-free voice-driven drawing application for people with motor impairments. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, 27–34. Tempe, AZ: ACM.

Henderson, V., S. Lee, H. Brashear, H. Hamilton, T. Starner, and S. Hamilton. 2005. Development of an American Sign Language game for deaf children. In *Proceedings of the 2005 Conference on Interaction Design and Children*, 79.

Hickey, D. T., A. C. H. Kindfield, P. Horwitz, and M. A. Christie. 2000. Integrating instruction, assessment, and evaluation in a technology-based genetics environment: The GenScope follow-up study. In *Proceedings of the International Conference of the Learning Sciences*, 6–13. Ann Arbor, MI.

Holdaway, D. 1979. *The Foundations of Literacy*. New York: Ashton Scholastic.

Hornof, A. J. 2008. Working with children with severe motor impairments as design partners. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 69–72. Chicago, IL: ACM.

Hornof, A. J., and A. Cavender. 2005. EyeDraw: Enabling children with severe motor impairments to draw with their eyes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 170.

Hourcade, J. P. 2002. *It's Too Small! Implications of Children's Developing Motor Skills on Graphical User Interfaces* (No. CS-TR-4425): University of Maryland Computer Science Department.

Hourcade, J. P., K. B. Perry, and A. Sharma. 2008. PointAssist: Helping four year olds point with ease. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 202–9.

Hoysniemi, J., P. Hamalainen, and L. Turkki. 2003. Using peer tutoring in evaluating the usability of a physically interactive computer game with children. *Interact Comput* 15:203–25.

Hutchinson, H., W. Mackay, B. Westerlund, B. B. Bederson, A. Druin, C. Plaisant et al. 2003. Technology probes: Inspiring design for and with families. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Fort Lauderdale, FL.

Inkpen, K. 1997. Three important research agendas for educational multimedia: Learning, children and gender. In *Proceedings of Educational Multimedia '97*, 521–6. Calgary, AB.

Inkpen, K. 2001. Drag-and-drop versus point-and-click: Mouse interaction styles for children. *ACM Trans Comput Hum Interact* 8(1):1–33.

Inkpen, K., S. Gribble, K. S. Booth, and M. Klawe. 1995. Give and take: Children collaborating on one computer. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 258–9. Denver, CO.

Ishii, H., and B. Ullmer. 1997. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI Conference on Human Factor in Computing Systems*, 234–41. New York: ACM.

Iversen, O. S. 2002. Designing with children: The video camera as an instrument of provocation. In *Proceedings of the Conference on Interaction Design and Children*. Eindhoven, The Netherlands.

Iversen, O. S., K. J. Kortbek, K. R. Nielsen, and L. Aagaard. 2007. Stepstone: An interactive floor application for hearing impaired children with a cochlear implant. In *Proceedings of the 6th International Conference on Interaction Design and Children*, 124.

Jensen, J. J., and M. B. Skov. 2005. A review of research methods in children's technology design. In *Proceedings of the Conference on Interaction Design and Children*. Boulder, CO.

Joiner, R., D. Messer, P. Light, and K. Littleton. 1998. It is best to point for young children: A comparison of children's pointing and dragging. *Comput Hum Behav* 14(3):513–29.

Jones, T. 1992. Recognition of animated icons by elementary-aged children. *Assoc Learn Technol J* 1(1):40–6.

Kail, R. 1991. Developmental changes in speed of processing during childhood and adolescence. *Psychol Bull* 109:490–501.

Kay, A. 1972. A personal computer for children of all ages. In *Proceedings of the ACM National Conference*. Boston, MA.

Kemp, J. A. M., and T. van Gelderen. 1996. Co-discovery exploration: An informal method for the iterative design of consumer products. In *Usability Evaluation in Industry*, ed. P. W. Jordan, B. Thomas, B. A. Weerdmeester, and I. L. McClelland, 139–46. London: Taylor & Francis Ltd.

Kolodner, J., and M. Guzdial. 1996. Effects with and of cscl: Tracking learning in a new paradigm. In *CSCL: Theory and Practice*, ed. T. Koschmann. Mahwah, NJ: Lawrence Erlbaum Associates.

Koschmann, T., ed. 1996. *CSCL: Theory and Practice*. Mahwah, NJ: Lawrence Erlbaum Associates.

Lamberty, K. K., and J. Kolodner. 2005. Camera talk: Making the camera a partial participant. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 839–48. Portland, OR.

Lenhart, A. 2007. Cyberbullying. In *Pew Internet & American Life Project*. http://pewinternet.org/Reports/2007/Cyberbullying.aspx (accessed November 1, 2011).

Leo, G. D., and G. Leroy. 2008. Smartphones to facilitate communication and improve social skills of children with severe autism spectrum disorder: Special education teachers as proxies. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 45–8. Cicago, IL: ACM.

Li, Q. 2006. Cyberbullying in schools: A research of gender differences. *Sch Psychol Int* 27(2):157.

Li, Q. 2007. New bottle but old wine: A research of cyberbullying in schools. *Comput Hum Behav* 23(4):1777–91. doi:10.1016/j.chb.2005.10.005.

MacFarlane, S., G. Sim, and M. Horton. 2005. Assessing usability and fun in educational software. In *Proceedings of the Conference on Interaction Design and Children*. Boulder, CO.

Markopoulos, P., and M. Bekker. 2002. How to compare usability testing methods with children participants. In *Proceedings of the Conference on Interaction Design and Children*, 153–8. Eindhoven, The Netherlands.

Martin, F., and M. Resnick. 1993. LEGO/Logo and electronic bricks: Creating a scienceland for children. In *Advanced Educational Technologies for Mathematics and Science*, ed. D. L. Ferguson, 61–90. Berlin, Heidelberg: Springer-Verlag.

Mayberry, R. 2007. When timing is everything: Age of first-language acquisition effects on second-language learning. *Appl Psycholinguist* 28(3).

McElligott, J., and L. van Leeuwen. 2004. Designing sound tools and toys for blind and visually impaired children. In *Proceedings of the Conference on Interaction Design and Children*, 65–72. College Park, MD.

Merryman, J., A. Tartaro, M. Arie, and J. Cassell. 2008. Designing virtual peers for assessment and intervention for children with autism. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 81–4. Chicago, IL: ACM.

Miller, C. 1998. Designing for kids: Infusions of life, kisses of death. In *Proceedings of the Proceedings of the Game Developers Conference*. Longbeach, CA.

Miller, L. T., and P. A. Vernon. 1997. Developmental changes in speed of information processing in young children. *Dev Psychol* 33(4):549–54.

Moeller, M. P. 2000. Early intervention and language development in children who are deaf and hard of hearing. *Pediatrics* 106(3):e43.

Newman, D., P. Griffin, and M. Cole. 1989. *The Construction Zone: Working for Cognitive Change in School*. Cambridge, England: Cambridge University Press.

Nielsen, J. 1993. *Usability Engineering*. London: Academic Press.

Nix, D., P. Fairweather, and B. Adams. 1998. Speech recognition, children, and reading. In *Proceedings of the SIGCHI Conference on Human Factors in Computings Systems*, 245–6. Los Angeles, CA.

O'Hare, E. A., and M. F. McTear. 1999. Speech recognition in the secondary school classroom: An exploratory study. *Comput Educ* 3(8):27–45.

Oosterholt, R., M. Kusano, and G. D. Vries. 1996. Interaction design and human factors support in the development of a personal communicator for children. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 450–7. Vancouver, Canada.

Ortega-Tudela, J. M., and C. J. Gomez-Ariza. 2006. Computer-assisted teaching and mathematical learning in Down Syndrome children. *J Comput Assist Learn* 22(4):298–307.

Papert, S. 1972. *On Making a Theorum for a Child*. Paper presented at the ACM National Conference, Boston, MA.

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Papert, S. 1987. Computer criticism vs. technocentric thinking. *Educ Res* 16(1):22–30.

Papert, S. 1992. *The Children's Machine.* NewYork: BasicBooks.

Patomäki, S., R. Raisamo, J. Salo, V. Pasto, and A. Hippula. 2004. Experiences on haptic interfaces for visually impaired young children. In *Proceedings of the 6th International Conference on Multimodal Interfaces*, 281–8.

Pea, R. 1984. On the cognitive effects of learning computer programming. *New Ideas Psychol* 2(2):137–68.

Pea, R. 1987. The aims of software criticism: Reply to Professor Papert. *Educ Res* 16:4–8.

Piaget, J. 1970. *Science of Education and the Psychology of the Child*. New York: Orion Press.

Piper, A. M., E. O'Brien, M. R. Morris, and T. Winograd. 2006. SIDES: A cooperative tabletop computer game for social skills development. In *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work*, 10.

Price, S, Y. Rogers, M. Scaife, D. Stanton, and H. Neale. 2003. Using 'tangibles' to promote novel forms of playful learning. *Interact Comput* 15(2):169–85.

Putnam, C., and L. Chong. 2008. Software and technologies designed for people with autism: What do users want? In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, 3–10. Halifax, NS: ACM.

Repenning, A., and L. E. Fahlen. 1993. Agentsheets: A tool for building domain-oriented visual programming environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 142–3. Amsterdam, The Netherlands.

Resnick, M. 1994. *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds*. Cambridge, MA: MIT Press.

Resnick, M., A. Bruckman, and F. Martin. 1996. Pianos not stereos: Creating computational construction kits. *Interactions* 3(5):40–50.

Resnick, M., J. Maloney, A. M. Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner et al. 2009. Scratch: Programming for everyone. *Commun ACM* 52(11):60–7.

Resnick, M., and N. Rusk. 1996. The computer clubhouse: Preparing for life in a digital world. *IBM Syst J* 35(3–4):431–40.

Resnick, M., and B. Silverman. 2005. Some reflections on designing construction kits for kids. In *Proceedings of the Conference on Interaction Design and Children*. Boulder, CO.

Revelle, G., and E. Reardon. 2009. Designing and testing mobile interfaces for children. In *Proceedings of the 8th International Conference on Interaction Design and Children*, 329–32. Como, Italy: ACM.

Rogers, Y., S. Price, G. Fitzpatrick, R. Fleck, E. Harris, H. Smith, C. Randell, H. Muller, C. O'Malley, and D. Stanton. Ambient wood: Designing new forms of digital augmentation for learning outdoors. In *Proceedings of the 2004 Conference on Interaction Design and Children: Building a Community*, 3–10. New York: ACM.

Rubin, J. 1994. *Handbook of Usability Testing*. New York: John Wiley and Sons, Inc.

Rusk, N., M. Resnick, R. Berg, and M. Pezalla-Granlund. 2008. New pathways into robotics: Strategies for broadening participation. *J Sci Educ Technol* 17(1):59–69.

Sánchez, J., and M. Sáenz. 2005. Developing mathematics skills through audio interfaces. In *Proceedings of 11th International Conference on Human-Computer Interaction, HCI*, 22–7.

Sánchez, J., M. Sáenz, and M. Ripoll. 2009. Usability of a multimodal videogame to improve navigation skills for blind children. In *Proceedings of the 11th International ACM SIGACCESS Conference on Computers and Accessibility*, 35–42. Pittsburgh, PA: ACM.

Scaife, M., and Y. Rogers. 1999. Kids as informants: Telling us what we didn't know or confirming what we knew already? In *The Design of Children's Technology*, ed. A. Druin, 27–50. San Francisco, CA: Morgan Kaufmann Publishers, Inc.

Scaife, M., Y. Rogers, F. Aldrich, and M. Davies. 1997. Designing for or designing with? Informant design for interactive learning environments. In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*. Atlanta, GA.

Scardamalia, M., and C. Bereiter. 1994. Computer support for knowledge-building communities. *J Learn Sci* 3(3):265–83.

Schneider, K. G. 1996. Children and information visualization technologies. *Interactions* 3(5):68–73.

Schneider, K. G. 1997. *The Internet Filter Assessment Project (TIFAP)*, [Website]. http://www.bluehighways.com/tifap/learn.htm.

Schuler, D., and A. Namioka, ed. 1993. *Participatory Design: Principles and Practices*. Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Skinner, B. F. 1968. *The Technology of Teaching*. New York: Appleton-Century-Crofts.

Soloway, E., M. Guzdial, and K. E. Hay. 1994. Learner-centered design: The challenge for HCI in the 21st century. *Interactions* 1(1):36–48.

Soloway, E., S. L. Jackson, J. Klein, C. Quintana, J. Reed, J. Spitulnik, S. J. Stratford et al. 1996. Learning theory in practice: Case studies of learner-centered design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Vancouver, Canada.

Songer, N. B. 1996. Exploring learning opportunities in coordinated network-enhanced classrooms: A case of kids as global scientists. *J Learn Sci* 5(4):297–327.

Stewart, J., E. M. Raybourn, B. Bederson, and A. Druin. 1998. When two hands are better than one: Enhancing collaboration using single display groupware. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Los Angeles, CA.

Stiehl, W. D., J. K. Lee, C. Breazeal, M. Nalin, A. Morandi, and A. Sanna. 2009. The huggable: A platform for research in robotic companions for pediatric care. In *Proceedings of the 8th International Conference on Interaction Design and Children*, 317–20. Como, Italy: ACM.

Strommen, E. 1994. Children's use of mouse-based interfaces to control virtual travel. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: Celebrating Interdependence*. Boston, MA.

Strommen, E. 1998. When the interface is a talking dinosaur: Learning across media with actimates barney. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Los Angeles, CA.

Strommen, E., and K. Alexander. 1999. Emotional interfaces for interactive aardvarks: Designing affect into social interfaces for children. In *Proceedings of the ACM Conference on Human Factors in Computing Systems: The CHI is the limit*. Pittsburgh, PA.

Tarrin, N., G. Petit, and D. Chêne. 2006. Network force-feedback applications for hospitalized children in sterile room. In *Proceedings of the 2006 Conference on Interaction Design and Children*, 157–60. Tampere, Finland: ACM.

Taylor, R. P., ed. 1980. *The Computer In the School, Tutor, Tool, Tutee*. New York: Teachers College Press.

Thomas, J. R. 1980. Acquisition of motor skills: Information processing differences between children and adults. *Res Q Exerc Sport* 51(1):158–73.

Tinker, R. 1993. *Thinking About Science*. Concord, MA: The Concord Consortium.

United Nations. 2006. Convention on the rights of persons with disabilities. http://www.un.org/disabilities/convention/facts.shtml (accessed May 18, 2010).

U.S. Department of Education, National Center for Education Statistics. 2011. *Digest of Education Statistics* (NCES 2011-015), Chapter 2.

van Kesteren, I. E., M. M. Bekker, A. P. O. S. Vermeeren, and P. A. Lloyd. 2003. Assessing usability evaluation methods on their effectiveness to elicit verbal comment from children subjects. In *Proceedings of the Conference on Interaction Design and Children*. Preston, UK.

Walker, D. F. 1987. Logo needs research: A response to Professor Papert's paper. *Educ Res* 9–11.

Walker, B. N., M. T. Godfrey, J. E. Orlosky, C. Bruce, and J. Sanford. 2006. Aquarium sonification: Soundscapes for accessible dynamic informal learning environments. In *Proceedings of the 12th International Conference on Auditory Display*, 238–41.

Westeyn, T. L., J. A. Kientz, T. E. Starner, and G. D. Abowd. 2008. Designing toys with automatic play characterization for supporting the assessment of a child's development. In *Proceedings of the 7th International Conference on Interaction Design and Children*, 89–92. Chicago, IL: ACM.

Winograd, T. 1996. Profile: Kid pix. In *Bringing Design to Software*, ed. T. Winograd, 58–61. New York: ACM Press.

37 Information Technology for Communication and Cognitive Support

Alan F. Newell, Alex Carmichael, Peter Gregor, Norman Alm, Annalu Waller, Vicki L. Hanson, Graham Pullin, and Jesse Hoey

CONTENTS

37.1 SUMMARY

Well-designed communication and information technology systems can substantially enhance the quality of life and independence for those with cognitive dysfunction, including elderly people. They can do the following:

- Allow people to retain a high level of independence and control over their lives
- Provide appropriate levels of monitoring and supervision of at-risk people, without violating privacy
- Keep people intellectually and physically active
- Provide communication methods to reduce social isolation and foster social inclusion

It is not easy to understate the importance of designers considering the effects of cognitive impairment on users. For example, access to the Internet for people with disabilities is often thought to be synonymous with access for people with visual impairment, but people with visual impairment form only a small percentage of the disabled population. Ogozalec (1997, p. 65) pointed out that, if current trends continue in the United States, by 2030, one-fifth of the population will be over 65 years of age and commented, "It is difficult to categorize and draw conclusions about 'the elderly,' since they comprise such a diverse and heterogeneous population." This diversity, particularly of cognitive function, needs to be taken into account if we are to make software and the Internet available to as large a percentage of the population as possible.

To illustrate the many challenges of cognitive impairment, this chapter will describe the major types of cognitive impairment including the effects of aging, and the effects these can have on daily living.

The increasing power and decreasing size of information technology, along with its capacity to provide communication as well as computation and storage, offers the possibility of quite sophisticated help for cognitive impairments. Computers have the potential to act as a scaffolding for cognitive tasks, taking over functions that have been affected by illness, accident, or aging. They can provide support for communication, both spoken and written, prompts for daily living, and entertainment and support systems for people with dementia.

The design and development of systems to support people with a variety of cognitive impairment will be illustrated by specific projects in which the authors have been involved. These examples do not cover all aspects of cognitive impairment, but illustrate an approach to developing assistive technology which has proved successful in a number of application areas.

This chapter will address the development of methodologies that are valuable for designers of information technology systems. These are relevant both to designers of specialist systems to support people with cognitive impairment and for designers of "main stream" systems—so that they are able to take into account the needs of people with cognitive impairments.

37.2 INTRODUCTION

Using technology to augment human cognitive capacity is not a new idea. In its widest sense, it includes all the tools and techniques that humans have developed to help support and/or enhance their cognitive abilities. The first cognitive function to be augmented was probably memory, and mnemonic methods help with this are still in wide use today. One of the most common (the method of loci) is to link what is to be remembered with a well-established memory structure that is easy to recall, such as the layout of a familiar city, or a narrative that has already been memorized. The introduction of written language extended people's cognitive abilities by allowing memories to be recorded externally. At the time, some feared this backwards step would allow people's memory powers to wither, but the overriding effect has been to free cognitive abilities that can further develop on the basis of the external support. This initial response to cognitive support systems is very common and continues today where information processing technologies have substantially extended the potential of most people's cognitive abilities. However, this technology has a similar potential to extend the cognitive abilities of those with some form of impairment or other limitation.

Particular strengths of computers as assistants for people with cognitive impairments include being consistent, tireless, and not becoming emotionally involved. In addition, multimedia and multimodal systems can provide a very rich interaction, which may be particularly advantageous for users with cognitive dysfunction. Such systems have great potential in addressing the problems of memory loss and the related difficulties presented by dementia. Communication systems that use synthetic speech, predictive programs that facilitate writing, and a range of nonlinguistic methods of communication, can be used by those with speech and language dysfunction caused by cognitive impairments.

37.3 COGNITIVE IMPAIRMENT

The use of the term cognitive impairment implies that two categories of human cognitive systems exist—impaired and unimpaired. However, this is not the case, although it can reasonably be stated that there are normal or average cognitive systems. The vast majority of experimental cognitive psychology literature relates to this normal system. In many contexts, this level of explanation is suitable for indicating of what most people are capable. It should always be borne in mind, however, that in real-world situations there is no marked distinction between that which is normal and that which is not. In other words, everyone has some limits to their cognitive abilities, which are considered normal. Within these bounds of normalcy, everyone will be better at some types of thinking than other types (e.g., math versus language) and often these differences can be relatively extreme. Most of the time, these weaknesses can (to some extent) be compensated for by the strengths, thus general levels of cognitive performance remain effectively normal. However, for a significant minority of people weakness can effectively outweigh strengths and this will generally be considered impairment. Some have a highly specific impairment, some have more diffuse problems, and some experience interrelated constellations of impairments which can be exacerbated by associated physiological impairment. In addition, the cognitive abilities of any one human being will change over longer and shorter periods. Aging can have substantial effects on cognitive ability, which is particularly marked in some age-related conditions, such as dementia.

For ease of exposition, the forms of cognitive impairment identified and described below will in the main refer to general categories. It should be noted, however, that all these categories lie somewhere on a continuum and, while they are delineated on the basis of educational and clinical or medical criteria, such cutoff points are relatively arbitrary in the context of the wide variability of cognitive ability across the population. It is also worth noting that within the context of normal cognitive systems, there is significant diversity among people with regard to differential preferences for types of material and ways of approaching and processing information. For example, some people may be considered primarily verbal and tend to excel in language-based tasks, relative to those considered visuo/spatial (e.g., Lohman 2000). Thus, many of the types of impairment addressed below can be construed to an important extent as the extremities of normal diversity.

37.3.1 Intelligence Quotient

The most widely known dimension of general cognitive ability is probably intelligence. Scientific investigation of this dimension has a controversial past, and many aspects of this are beyond the scope of this chapter (for a more comprehensive account, see Gould [1997]). One underlying reason for such controversy is that the word intelligence has a rather nebulous definition. In day-to-day usage, this is rarely problematic, but the differences between scientific and lay definitions can cause misunderstanding (e.g., Sternberg 2000). Such misunderstanding can lead to controversy as most definitions of intelligence include connotations that are considered socially important and, thus, can often be highly emotive. Despite these difficulties, the investigation of intelligence has provided many insights into a wide range of more particular cognitive abilities, and has developed methods for quantifying general intellectual ability such as the various forms of intelligence quotient (IQ) tests. Again, controversy has surrounded the use of these tests over the years (see Gould [1997]; Kaufman [2000]), but such tests have been widely used and are accepted as a general benchmark of a person's intellectual capability.

An IQ score of 100 is, by definition, normal with about 50% of the population scoring above and 50% scoring below, but it should be noted that elderly people are not generally included in the standardization of these scores. Approximately 50% of the population is considered within the bounds of normality and deviate either side of 100 by no more than 10 points. The nonnormal 50% are distributed approximately evenly above and below this band. Thus, about a quarter of the (nonelderly) population fall below the level of what is considered normal. Although the terminology varies across cultures and over time, around 20% of the population have IQ scores between 75 and 90 and would generally be classified as slow learners. The final 5% will generally have very special needs that overall are best addressed on an individual basis (Kaluger and Kolson 1987). Further to this and as an example of the emotive connotations associated with the issue of "intelligence," it is worth noting that the first official classification scheme (see Detterman, Gabriel, and Ruthsatz [2000]) associated with IQ tests further broke down this latter 5%. These classifications were moron (IQ 50–75), imbecile (IQ 25–50), and idiot (IQ, <25), terms which today would be considered wholly unacceptable as a description of anyone with a cognitive impairment. The above-normal segment of the population has received much less research interest, will tend to be high achievers, and are only mentioned here to highlight that "normal" *could* be construed as "impaired."

An IQ score reflects a person's intellectual ability as a whole. A low score may be due to the whole system functioning at a suboptimal level, but a similar result can also be due to one or more component abilities being impaired. There are many tests of IQ, some of which give an indication of this while others do not. Some IQ tests are explicitly broken down into subtests that reflect the relative levels of ability in the component cognitive abilities, such as the verbal and visuo/spatial abilities mentioned above. The more common forms of cognitive impairment are described below in the context of a brief overview of the cognitive system.

For any information in the outside world to enter the cognitive system, it must first be detected and transmitted by the sensory apparatus. In an important sense, this is not simply the start of the process because aspects of attention will influence what is and is not detected/transmitted, and, to a certain extent, how. Basic perceptual processing creates a sensory specific representation of the stimulus event. Streams of such stimulus events are summated into meaningful cognitive entities (e.g., strokes on a page recognized as letters and numerals are summated into a name and telephone number). These will then be either: passed immediately to short-term memory (STM), further processed by working memory

(WM), or rehearsed for maintenance in STM or for encoding into longer term storage. For example, rehearsal could refer to rote rehearsal of a telephone number between reading and dialing it or to more elaborate processing to associate it with relevant extant memories to improve the chance of subsequent recall (e.g., method of loci, mentioned above).

Output from the cognitive system will generally be initiated in response to some form of external stimulus, or probe, by accessing extant memories relevant to the probe using executive processes to organize them in a task-relevant way and then producing a response. Output of this kind has been most commonly studied with the use of memory tests. This minimizes the influence of intellectual processing (problem solving, etc.) per se, and emphasizes the registration, rehearsal, and encoding of information, the effects of decay, interference and other forms of forgetting, and the effectiveness of different cues (probes) in eliciting specific memories (e.g., recall vs. recognition).

Virtually all aspects of the processing outlined above are shaped by attention, and it is important to note that, regardless of impairment, while we all have some control over attention, it can also be the case that attention can have some control over us. That is, we can utilize attention to focus on searching a list for a particular telephone number while ignoring the chatter of people around us. Having read the number, however, our attention can exert its own control if someone calls our name and asks if we have made that call yet. Despite our best efforts at rehearsal, it is likely that our attention will be grabbed by our name, and the ensuing question and this brief distraction can be enough to lose the information from temporary storage.

In general terms, mild to moderate global cognitive impairment will be associated with decrements in efficiency across most of the processing stages outlined above and in aspects of the utilization of attention. The following will describe some of the main decrements in cognitive ability related to interacting with computer-type systems.

37.3.2 Attention Deficit

Many of the constraints imposed by cognitive impairment can be further shaped by decrements in various aspects of attention. One major aspect of this is generally referred to as "selective attention," which allows people to focus on salient aspects of a task and at the same time helps them actively to ignore irrelevant aspects. The efficiency of selective attention is markedly diminished in most forms of cognitive impairment. This factor further supports the recommendation to present the user with just one thing at a time, which will avoid them erroneously devoting time and cognitive resources to processing irrelevant information. Similarly, if the nature of the interaction requires the user to attend to some critical information at a particular time/location, appropriately obvious highlighting should be used to attract the users' attention. These issues become emphasized in situations where selective attention must be maintained over periods of more than just a few minutes.

Another aspect of attention known to be less efficient in cognitive impairment is referred to as "divided attention." In general, this refers to the ability to allocate cognitive resources appropriately when trying to do two or more distinct cognitive tasks or distinct portions of the same task at the same time.

37.3.3 Autism

Another distinct form of global cognitive impairment is autism, including a set of rarer but related syndromes (Kaluger and Kolson 1987). The precise causes of autism are not clearly understood. Briefly stated, it is a general neurological disorder that impacts the normal development of the brain particularly in relation to social interaction and communication skills. Its effects will usually become apparent within the first 3 years of life. People with autism typically have difficulties in verbal and nonverbal communication and social interactions. The disorder makes it hard for them to communicate with others and relate to the outside world, they also tend to have relatively low IQ scores. Closely related to autism is Asperger syndrome. People with Asperger's experience similar social communication difficulties, but generally demonstrate a normal IQ. Further to this, there are several generally similar conditions, some of which have varying physical and behavioral elements associated with them. These come under the collective heading of pervasive developmental disorders and all tend to produce difficulties with communication. An important element of these social communication difficulties in the context of the present chapter is an inability to grasp the implications of metaphorical or idiomatic language. Similar effects occur in dementia but in autism tend to be more profound. There is some evidence that people with autism or Asperger syndrome are more able to communicate with computers than with people, or with people via computers, rather than face-to-face, and thus properly designed computer systems may have potential for assisting such user groups.

37.3.4 Aphasia and Complex Communication Needs

Cognitive limitations can have a major impact on the ability to encode and decode traditional orthography. People with acquired cognitive impairments (e.g., aphasia resulting from a stroke or cerebral vascular accident) can experience varying degrees of difficulty with both expressive and receptive language. A person with aphasia may have slight word-finding problems through to more pervasive problems in understanding spoken language. Although some individuals may retain some literacy skills, damage to the language processing centers will usually also affect symbolic representations of language.

Individuals with congenital language and/or intellectual disabilities (e.g., developmental delay and Down syndrome) may never become literate. The physical inability to speak (e.g., dysarthria resulting from cerebral palsy) may also impact literacy learning, as basic skills required for reading and writing (e.g., phonemic awareness) may be absent. Children with complex communication needs (CCN), for example, as a result of cerebral palsy, have a combination of physical, speech, and/or intellectual impairments that impact on the development of speech and language.

37.3.5 Cognitive Effects of Aging

It is well known that aging brings about changes in a person's abilities. Perceptual (vision and hearing) and motor declines, for example, are well documented and a number of best practices and design guidelines exist that help designers with resulting interface needs (see, e.g., Arch [2008]; Coyne and Nielson [2002]; Hanson et al. [2009]; Kurniawan [2009]; Morrell et al. [2002]). In contrast, there is less understanding about cognitive declines that accompany aging, and certainly less consensus on ways to support older users who find computing challenging due to some of these declines.

In healthy aging, there are a number of cognitive activities that can affect ability to use technology (Chin and Fu 2010; Czaja et al. 2010; Fairweather 2008; Hanson 2009; Park 1992; Rabbitt 1993; Sayago, Camacho, and Blat 2009). The rate and degree of cognitive decline vary from person to person, but, as a general statement, cognitive declines can begin in middle age and continue throughout the rest of one's life. A distinction between *crystallized intelligence* and *fluid intelligence* is often made to help understand the complexity of cognitive abilities and aging. These two terms encapsulate aspects of intelligence that have different patterns of decline with age. Crystallized intelligence refers to verbal ability and general knowledge. It characteristically remains intact throughout a person's lifetime. In fact, there is generally a slight increase in crystallized intelligence abilities until old age, at which point slight declines begin. In contrast, fluid intelligence refers to a set of cognitive abilities that decline steadily (albeit at different rates for different individuals) beginning in middle age. Fluid intelligence includes such skills as STM, speed of processing, and problem-solving ability. Critically, fluid intelligence abilities are exactly those needed to be able to learn new computing skills (Czaja et al. 2006) and successfully perform technology tasks such as web searching (Czaja et al. 2010). Rabbitt (1993) has shown that, in many circumstances, relevant accumulated knowledge can ameliorate declines in fluid ability. However, in other circumstances the opposite can be the case, wherein a well-learned, but essentially inappropriate, strategy can put a relatively greater burden on the associated fluid abilities.

37.3.6 Dementia

Various forms of dementia can exaggerate the relatively mild effects of normal aging on the cognitive system. At the age of 60 years, about 1% of the population is diagnosed with dementia. This percentage approximately doubles for every subsequent 5-year-age band (e.g., 4% at 70 years and 16% at 80 years; Bäckman et al. 2000). Of the elderly population with dementia, Alzheimer's disease accounts for about 60%, depending on the diagnostic criteria used and a further 25% is vascular dementia (e.g., related to circulatory problems). Most of the vascular dementias are referred to as "multi-infarct dementias" and tend to be caused by series of mini-strokes, and thus, they tend to have more diffuse and less predictable effects on ability than a major stroke. The remaining 15% of dementias are made up of various relatively rare conditions (Bäckman et al. 2000).

Regardless of the various causes and effects, all forms of dementia involve damage to the brain, such damage being more or less widespread, affecting cortical and/or subcortical areas. In general, damage to the cortex results in cognitive/perceptual impairment, whereas damage to the subcortical areas is more related to physical impairment. However, there are a number of well-known problems related to the diagnosis of dementia. Two of these are relevant here. The first involves the grey area between the worst effects of normal aging and the initial effects of pathological aging at the onset of dementia. The second is that the effects of depression in later life can closely mimic those of dementia. These additional complexities further expand the overall diversity of cognitive impairment in relation to human-interface design, both in regard to the general level of ability and in the variation of that level over periods of days, weeks, months, and even years. The convolution of this situation is further added to by the effects of a relatively greater probability of ill health among older people. It is estimated that around 80% of those aged 65 and over have at least one chronic illness and that many will have more than one. In addition to the effects of health per se, there is also potential for cognitive ability to be affected by a variety of medication and by interactions between different medicines.

Despite the above, some systematic changes associated with extreme old age and dementia are relevant to human-interface design. In general, the first ability to deteriorate in dementia, particularly with Alzheimer's disease, is episodic as distinct from semantic memory. Episodic memory is memory for events, usually from the viewpoint of personal experience, rather than for facts. That is, remembering, "X is the capital of Y" is the product of semantic memory, whereas remembering when and where you were while you were reading "X is the capital of Y" is the product of episodic memory. This generalized decrement in episodic memory may be related to findings in normal aging research such as disproportionate decrements in source memory (e.g., specifically remembering where an item was rather than what it was) and to prospective memory (e.g., remembering to do something in the future). These changes have important implications for successful navigation in interactive systems. For example, keeping track of where you have just been is often an important prompt to where you are going now.

37.3.7 Dyslexia

Dyslexia is a neurological problem that manifests itself as a language disorder (Dyslexia Research Institute 2010; National Center for Learning Disabilities 2007; Shaywitz et al. 1998).

It can present a number of language-related difficulties and definitions of it have varied. The British Dyslexia Association (2006) offered the following description:

> Dyslexia is best described as a combination of abilities and difficulties that affect the learning process in one or more of reading, spelling, and writing. Accompanying weaknesses may be identified in areas of speed of processing, short-term memory, sequencing and organization, auditory and/or visual perception, spoken language and motor skills. It is particularly related to mastering and using written language, which may include alphabetic, numeric and musical notation.

One of the main features of dyslexia is the individual nature of the disorder. The condition is not typically characterized by one single difficulty, but by a range of difficulties that will vary in combination and in intensity between individuals, giving rise to an enormous variation between individuals in the problems encountered. Each dyslexic person thus has a range of difficulties that need to be addressed differently from those of others. Dyslexia is an example of the need to design for dynamic diversity (see Section 37.12.4).

The wide-ranging characteristics of dyslexia provide a challenge for technological assistance, as a single approach will not be appropriate for the range of problems presented by the population of dyslexic people. Computer technology offers the opportunity to provide reading and writing systems that are highly configurable for each individual user. These systems, however, need to be based on an understanding of the problems that dyslexics have in reading and writing.

Some of the most commonly encountered problems are as follows (adapted from Willows, Kruk, and Corcos [1993]):

1. Number and letter recognition. One of the fundamental problems faced by dyslexics is the recognition of individual alphanumeric symbols. This is often seen when letters that are similar in shape, such as n and h and f and t, are confused. The problem is exacerbated with the introduction of uppercase letters. In addition, many dyslexic adults, who are capable of reading printed letters, have difficulty in reading cursive writing.
2. Letter reversals. Many dyslexics are prone to reversing letters, which results in a particular letter being interpreted as another letter. Examples of these characters are b, d, p, and q. This problem can result in poor word recognition with words containing reversal characters being substituted for other words, such as bad for dad.
3. Word recognition. As well as the substitution effect caused because of letter reversals, words that are similar in their outline shape (word contour) can be substituted by dyslexics. Typical examples of this problem are the words either and enter. Both words have the same start and finishing characters and this, allied with their similar word contours, make them candidates for being substituted for each other when they occur in the text.
4. Number, letter, and word recollection. Even if the ability to recognize numbers and letters is adequate, it can still prove difficult for a dyslexic individual to recall the actual form and shape of a character. Many dyslexics have so much difficulty recalling upper and lower case characters that they continue to print later in life. Similarly, poor visual memory means that dyslexics have little ability to distinguish whether or not a word looks right.
5. Spelling problems. Due to the problems discussed above, dyslexics can have great difficulty with spelling, and many dyslexics have very poor spelling. Much of the spelling of dyslexics appears to reflect a phonic strategy with words like *of* and *all* being spelled *ov* and *ohl*.
6. Punctuation recognition. As with characters, dyslexics appear to have difficulty recognizing punctuation marks.
7. Saccadic and fixation problems. Another problem that is found in many dyslexics is their lack of ability to follow text without losing their places. Many find it difficult to move from the end of one line to the beginning of the next and find themselves getting lost in the text.
8. Word additions and omissions. Dyslexics may add or remove words from a passage of text, apparently at random. This is manifested by words being omitted or duplicated, extra words being added, or word order being reversed or otherwise jumbled.
9. Poor comprehension. With the variety of errors caused by the factors described above, a dyslexic person may perceive a totally different (or impoverished) passage of text from the one that is actually in front of them. Dyslexics thus display poor comprehension skills due to text which they perceive being significantly different from the actual text.

37.4 INTERFACE DESIGN TO SUPPORT PEOPLE WITH COGNITIVE IMPAIRMENT

Specialist information technology has been developed to provide support for people with various types of cognitive impairment, and some such projects are described later. It is also important, however, to address the challenge of providing access to more mainstream technology for people with cognitive impairments. When designing or specifying mainstream technology, it is important to focus on the characteristics of such users and to be fully aware of the range of cognitive diversity, even among those without clinical dysfunction. This is rarely mentioned in human-interface design, where the cognitive diversity of the human race has not been the focus of much research. It is also important to consider the effects of age on cognitive function when designing information technology systems.

37.4.1 Speed

A key aspect of any intellectual task, in regard to interactive technology for people with mild or moderate global cognitive impairment, is speed (Salthouse 1991). Whatever level of performance a person can achieve in any given situation will be made worse if the task must be done under externally imposed time constraints whether actual or simply inferred by the user. Thus, wherever possible, the design of any interaction should allow every step to be carried out at the user's own pace. Attention to text layout can also benefit people with specific learning difficulties, such as those with dyslexia and people with limited literacy levels. Beyond this, clear text and presentation layout will always be worth considering carefully as such aspects have been found to benefit those who do not specifically need it, albeit to a less marked extent (Freudenthal 1999; Pirkl 1994).

37.4.2 Simplicity

Another key concept related to interface design for people with cognitive impairment is the avoidance of complexity. Complexity in interfaces can manifest itself in many different ways and at many different levels. A truly comprehensive coverage of this is beyond the scope of the present chapter, but some illustrative examples will be given to illustrate "cognitive accessibility."

Cognitive accessibility is more difficult to measure than sight and hearing accessibility, but is absolutely vital when designing for people with cognitive impairment. Simplicity supports cognitive inclusion but it is not easy to achieve. Particularly with information technology products, it is often easier to solve design challenges by providing more functionality, forgetting that this may confuse the user. Also many developers of so called "universal design" product interpret that to mean that a product or service should do everything that any individual could want, and that it should be all things to all people (Pullin 2009b). This can lead to a very complex final solution. It should be noted that ability to customize a user interface—with the intension of "simplifying it"—often makes the overall product even more complex. The goal of simplicity—the ability to use a product without thought—should be the goal of designers of products aimed at users with cognitive dysfunction, and this means paring down the functionality of the product as much as possible. In a sense, this is the antithesis of "user centered design" where the user is asked what they want in a product—designers need to be conscious of the "if it is offered why not say yes," response of users. The iPod Shuffle provides an example of the success of a simplified product. It can be argued that focus groups of users would be unlikely to have asked for such a minimal set of functions in a music player, but these devices have been a commercial success.

37.4.3 Language and Interfaces

The use of language in an interactive system should be given careful consideration, and the syntax and vocabulary should be kept as straightforward and commonplace as the context allows. This is particularly pertinent for any form of instruction. If the requirements of a particular stage of an interaction cannot be captured in a few simple concrete statements, then serious consideration should be given to redesigning the interaction itself. Similarly any on-screen display should be kept as uncluttered as practicable and, wherever possible, should present the user with only a single issue (menu, subject, decision, etc.) at any particular point in time. Similarly, but at a larger scale, progression through an interaction should be kept, wherever practicable, as linear as possible. That is, the user should only need to consider one thing at a time. Any requirement to deal with different decisions in parallel will markedly increase the possibility of errors and general user dissatisfaction (Detterman, Gabriel, and Ruthsatz 2000; Salthouse 1985).

As the designers of a system will have a comprehensive understanding of the functions of that system, however, they are unlikely to assess issues of complexity from the users' point of view, particularly that of a novice user. In addition, prescriptive checklists for avoiding complexity will ultimately be inadequate, as the optimum approach will always depend of the specifics of the task the interactive system is intended to support (Carmichael 1999). This is one of the main issues that highlights the importance of early and rigorous user involvement in the design of interactive systems, and is particularly important in the case of young designers developing interfaces for older users and those with cognitive impairment.

37.4.4 Design to Support Attention Deficit

In many software systems, the user is required to do more than one thing at a time, and this can simply demand more cognitive resources than are available. However, declines in the efficiency of divided attention can mean that, even if the tasks involved demand no more than the resources available, they may not be allocated appropriately. Interactive systems should be designed to relieve the user of this kind of burden. It is difficult to be prescriptive about suitable solutions to this problem, as the appropriate approach will depend on the specifics of the interaction involved, but some general concepts are useful to bear in mind. For example, the provision of some form of notepad function may be helpful for temporarily recording information for subsequent use (although great care is needed to ensure that the instantiation of such a function and its utilization does not put further cognitive load on the user). Another possibility is the provision of an overview of the task, which could show or remind the user where they are and what they have and have not done so far.

37.4.5 Design to Support Older People with Memory Loss

Limitations in memory particularly affect older people. Thus, wherever practicable, interactive systems should be designed to take the burden of memory off the user, for example, by judicious use of prompts and reminders. Also, careful consideration of the steps in an interaction and the way they are presented to the user can help mitigate the most common problem of deficient STM and WM. Even with the best design efforts, however, such problems are likely to make users with cognitive impairment relatively error prone. It is thus essential to ensure that the interactive system allows for error correction in an easy to use form. To ensure that the user spots such errors, the system should provide feedback regarding user actions and where appropriate elicit active confirmation from them.

Difficulties with complex page navigation are interpretable in terms of STM and processing changes. Clear and consistent on-screen layout and navigation, therefore, are hallmarks of good design for older adults (Arch 2008; Carmichael 1999; Charness and Bosman 1994; Coyne and Nielson 2002; Holt and Morrell 2002). In the design of web pages, consistent navigation of pages within a site and clearly structured information can reduce problems. Providing feedback about the entire sequence of a multistep event, such as when making online purchases could prove beneficial. Searching can be improved for older adults by nonhierarchical interfaces. Difficulties with browser basics such as the back, history, bookmarks, and search can all be understood as complex activities that tax limited cognitive systems. Better-supported information about visited sites and searched sites is crucial for older users. With interactive systems, other specific remedies to support users have been suggested. For example, judicious use of prompts have been shown to be helpful (Zajicek 2003).

In addition to memory problems, other fluid intelligence abilities that can cause difficulties for older technology users are visuo/spatial, perceptual, and speed of processing abilities. Decline in such abilities can cause difficulty with decoding layouts and utilizing any inherent organization of websites. Difficulties of navigation are exacerbated, particularly in web 2.0 content, by dynamic changes ("change blindness" being particularly strong for older adults), difficulty in identifying clickable areas, and lack of help for ever-changing content.

As declines for older people are often in more than one area, their combination can make accessibility more challenging than for users having only a cognitive limitation. Consider, for example, an older adult who has STM loss, as well as failing eyesight. Trying to learn the commands needed for specialized software to see or have on-screen material read aloud (such as with a screen readers) presents a challenge (Zajicek 2007). Specialized software can present a double burden for learning: both the desired application as well as the specialized software must be learned at the same time. Given problems with STM, the learning task can be significantly more difficult. Understanding the complexity of problems experienced by older adults and how these problems interact, and the "dynamic diversity" of user needs (Gregor, Newell, and Zajicek 2002), is critical to understanding the needs of older adults.

37.5 COGNITIVE SCAFFOLDING AND PROMPTS FOR COMMUNICATION

Augmentative and alternative communication (AAC) are systems which facilitate communication by nonspeaking people. These normally use speech synthesis technology to speak text which is inputted by the user. For severely physically impaired nonspeaking people, even with current speech output technology, however, speaking rates of 2–10 wpm are common, whereas unimpaired speech proceeds at 150–200 wpm.

In an attempt to improve this disparity, some progress has been made by using computers to replace or augment some of the cognitive aspects of communication. One sequence of AAC research projects took as its starting point the improvement of communication systems for physically impaired nonspeaking people. It became apparent that this could be done very effectively by developing models of the cognitive tasks involved in communication. This research has now spawned a new area of development, which is cognitive support for people with dementia, where communicative impairment is just part of their range of difficulties (see Section 37.8).

Although the cognitive processes underlying language use are incompletely understood, a number of theories that attempt to explain language use have been utilized to improve the functionality of communication systems for nonspeaking people. This approach to the problem usually involves taking a sociolinguistic view of language. Instead of focusing on the building blocks, or taking a bottom-up approach, the interaction as a whole is analyzed, paying attention to its goals, or taking a top-down approach to the communication. This may well be a realistic simulation of the natural process, since the production of speech by an unimpaired speaker occurs at such a rate that conscious processing and controlling of the speech at a microlevel is not possible. In common with other learned skills, speech is produced to some extent, automatically, with the speaker being aware of giving high-level instructions to the speech production system, but leaving the details of its implementation to the system.

The nonconscious control of much of speech production has been modeled in the CHAT (conversation helped by automatic talk) prototype (Alm, Arnott, and Newell 1992). This produced quick greeting, farewell, and feedback remarks by giving the user semiautomatic control of what form the remarks would take, within parameters that the user had previously selected. This mimicked the phenomenon of a speaker responding automatically to greetings and other commonly occurring speech routines, without giving the process any detailed thought.

The CHAT-like conversation described illustrates an attempt to achieve a particular communicative goal, achieving social closeness by observing social etiquette. Some recent research efforts have been directed at finding ways to incorporate large chunks of text into an augmented conversation, to help users carry out topic discussion. This has been driven by the observation that a great deal of everyday discourse is reusable in multiple contexts.

Much of this type of discourse takes the form of conversational narratives. Research into the conversational narrative at a sociolinguistic level indicates several interesting characteristics. These include the way in which narratives are told and to whom they are told. For example, a recent event is told repeatedly for a limited time to most people with whom the speaker has contact. As the event recedes in history, the narrative is retold when it is relevant to the topic of conversation. The length of the narrative depends on its age (the older a story is, the more embellished it can become, particularly if it has previously gone down well) and the time available within the conversation. The version of the story (the sequence of events may remain the same while the details or embellishments of a story can differ) depends on factors such as the conversational context and other interlocutors present.

One of the ways to make the retrieval of text chunks easier is to anticipate the chunks that the user may want to use. This has been achieved by modeling conversational narratives using techniques from the fields of artificial intelligence and computational linguistics (Waller 1992). The prototypes developed constantly adapt to the users' language use, thus mirroring the user's perception of where conversation items are stored. In this way, the system adapts to the way the user thinks instead of having the user learn a new retrieval system.

One of the arguments against using such prediction was raised when word prediction systems were first developed in the early 1980s. Therapists and teachers were concerned that nonspeaking people, especially children, would select what was offered on the screen rather than what they originally wanted to say. Although this may happen, research into predictive systems applied to writing suggests that they may carry over the help they offer and have a wider effect on the users' ability development. Some of this research reports an increase in written output by reluctant writers and people with spelling problems (Newell et al. 1992). A general improvement in spelling has also been noted. Children with language dysfunction and/or learning disabilities have shown improvement in text composition (Newell, Booth, and Beattie 1991). This research is in the writing domain, but the results suggest that predictive systems can offer assistance without becoming mere substitutes for creative expression.

Also, in unimpaired conversation speakers often change direction in their communication depending on chance occurrences, or on the sudden recollection of a point they would like to include. Thus, there is a degree of opportunism in all conversations. Another argument in favor of offering predicted phrases and sequences is that the current situation for most augmented communicators is that their conversations tend to be quite sparse, with control tending to reside with unaided speaking partners. If it is not possible to go boating on the lake, easily going off in any direction you please, is it not preferable to build a boardwalk out over the water than to stay on the shore?

One of the motivations to improve communication systems for nonspeaking people is the fact that they are commonly perceived by people who do not know them as being less intellectually capable than they actually are. It is often reported by nonspeaking people that they are considered unintelligent or immature by strangers. The issue of perceived communicative competence is one that needs increased attention (McKinlay 1991).

Related to this, an interesting finding emerged from work in which one of the authors was involved. Here, a prototype communication system was used to evaluate listeners' impressions of the content of computer-aided communication based on prestored texts, compared with naturally occurring dialogues. The nonspeaking user was able to use only prestored texts to conduct the conversations. Most of the text was material about one subject (holidays). A number of rapidly accessible comments and quick feedback remarks were also available. The unaided conversations were between pairs of normally speaking volunteers who were asked to converse together on the topic of holidays. Transcripts of randomly sampled sections of the conversations and audio recordings of reenactments of the samples with pauses removed were rated for social competence on a six-item scale (coefficient alpha 5 0.83) by 24 judges. The content of the computer-aided conversations was rated significantly higher than that of the unaided samples ($p < .001$). The judges also rated the individual contributions of the computer-aided communicator and the unaided partners on how socially worthwhile and involving these appeared. There was no significant difference between the ratings of their respective contributions ($p > .05$; Todman, Elder, and Alm 1995).

This finding came as something of a surprise to the researchers, since the original purpose had been to establish whether conversations using prestored material would simply be able to equal naturally occurring conversations in quality of content. Of course, the pauses in actual computer-aided conversation (removed in the above analysis) do have an effect on listeners' impressions of the quality of the communication, but this finding is of interest, since it suggests that, in some ways, augmented communication could have an edge over naturally occurring talk. A plausible explanation for this finding is that naturally occurring talk is full of high-speed dysfluencies, mistakes, substitutions, and other messy features that listeners tend to discount with their abilities to infer what the speaker is intending to say. Prestored material is by its nature selected because it may be of particular interest, and it is expressed more carefully than quick flowing talk, and thus may appear more orderly and dense with meaning than natural talk.

In addition to conversational narratives, another common structure in everyday communication is the script, particularly where the speaker is undertaking some sort of transaction. Scripts may be a good basis for organizing prestored utterances to attempt to overcome the problem of memory load when operating a complex communication system based on a large amount of prestored material. Users' memory load can be reduced by making use of their existing long-term memories to help them locate and select appropriate utterances from the communication system. Schank and Abelson (1977) proposed a theory that people remember frequently encountered situations in structures in long-term memory, which they termed scripts. A script captures the essence of a stereotypical situation and allows people to make sense of what is happening in a particular situation and to predict what will happen next. Other research (e.g., Vanderheiden et al. 1996) has shown the potential that similar script-based techniques offer to this field.

An initial experiment was devised (Alm, Morrison, and Arnott 1995) to investigate the potential of a script-based approach to transactional interactions with a communication system, and a prototype system was developed to facilitate this experiment. The aim was to ascertain whether or not a transactional interaction could be conducted using a script-based communication system. It was decided to simulate a particular transactional interaction that could reasonably be expected to follow a predictable sequence of events, for example, one that would be amenable to the script approach, to find out whether a computer-based script could enable a successful interaction.

The transaction chosen for the experiment was that of arranging the repair of a household appliance over the telephone. Although the script interface was a relatively simple one devised for the purpose of this experiment, it was successful in facilitating the interaction, and produced a significant saving in the amount of physical effort required.

To take this work further, a large-scale project was undertaken to incorporate scripts into a more widely usable device. The user interface of this system is made up of three main components as follows: (1) scripts, (2) rapidly produced speech acts, and (3) a unique text facility. The scripts component is used in the discussion phase of a conversation and consists of a set of scripts with which the user can interact. The rapid speech act component contains high-frequency utterances used in the opening and closing portions of a conversation and in giving feedback and consists of groups of speech-act buttons. This facility is based on previous work with CHAT. The unique text component is used when no appropriate prestored utterances are available and consists of a virtual keyboard, a word prediction mechanism, and a notebook facility.

To provide access to a set of scripts, an interface was devised that involved a pictorial representation of the scenes in the script. The pictorial approach was taken to give users easier access to the stored material and to assist users with varying levels of literacy skills. In this interface, scripts are presented to the user as a sequence of cartoon-style scenes. The scenes give the user an indication of the subject matter and purpose of the script and assist the user to assess quickly if the script is appropriate for current needs. Each scene is populated with realistic objects chosen to represent the conversation tasks that can be performed. Thus, the user receives a pictorial overview of the script, what happens in it, and what options are available. This assists the user to see quickly what the script will be able to do in the context of the current conversation. An example of the interface for the system is shown in Figure 37.1, which shows a scene within the doctor script.

Research into picture recognition and memory structures has demonstrated that groups of objects organized into realistic scenes corresponding to stereotypical situations better assist recognition and memory compared with groups of arbitrarily placed objects (Mandler and Parker 1976; Mandler 1984). The scene-based interface using a realistic arrangement of objects within a scene was therefore chosen to facilitate recognition and remembering by the user and thus reduce the cognitive load required to locate suitable objects during a conversation.

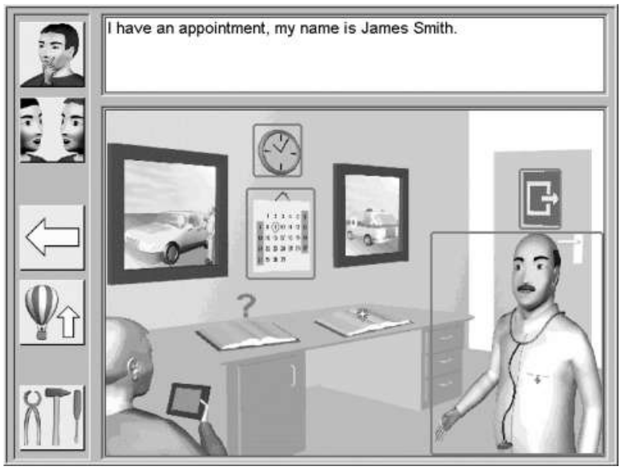

I have an appointment, my name is James Smith.

FIGURE 37.1 The script system user interface showing a scene from the doctor script.

As it would be impractical to provide scripts for every conceivable situation, it was decided to provide users with a limited number of scripts together with an authoring package with which they can develop their own custom scripts with help from their therapists.

A text preview and display box appears at the top of the user interface. The main interface area (bottom right) contains the scene image. The function buttons on the left side of the interface are, from top to bottom: “I’m listening” rapid speech-act button; button to access the main rapid speech-act interface; scene navigation backtrack to previous scenes button; scene navigation overview button; and tool button to access the notepad and additional system control facilities.

It was initially decided to develop six complete scripts. These were chosen after discussions with a user advisory group about situations in which they found difficulty communicating. The scripts developed were at the doctor, at the restaurant, going shopping, activities of daily living (ADL), on the telephone, meeting someone new, and talking about emotions.

The system uses the script to guide the user through a dialogue. There is a prediction mechanism, which predicts the next most probable stage in the dialogue that the user will need (based on the script), so the user can usually follow a predicted path through a conversation. This prediction mechanism monitors the sequences of objects selected and uses this information to modify future predictions.

Despite encouraging results with prototypes, the phrase-storage approach to social communication set out above has not gained widespread acceptance by users. The reasons for this are complex, but include: the relative inflexibility of prestored material; the costs associated with authoring the material and keeping the material up-to-date; and the vastly different nature of the approach and different training requirements necessary to achieve success.

37.6 SUPPORT FOR PEOPLE WITH COMPLEX COMMUNICATION NEEDS

AAC systems can be used when a physical disability alone precludes speech. Such systems can also provide methods to compensate (either temporarily or permanently) for the impairment and disability patterns of individuals with CCN. Some individuals will learn to read and write using traditional orthography (spelling) in combination with nonverbal means (e.g., facial expression) to communicate, and many AAC interfaces use traditional orthography (the written word). However, children with CCN are at great risk of not acquiring literacy. Visual images, photographs, and drawings are used to provide augmentative and alternative ways to access communication. Such images can be used to enhance text-based interfaces. In addition, sets of symbols (e.g., the Picture Communication System, PCS, Rebus, and semantic-based writing systems such as Blissymbolics), can be used as an alternative to text (Beukelman and Mirenda 1998; Wilson 2003). The type of picture, symbol, or graphic used will depend on the iconicity (ease of recognition), transparency (guessability), opaqueness (logic organization), and learnability of the image. For instance, a photograph of a house may be transparent—for example, it is easily recognizable—while a Blissymbol (see below) representing the emotion of happy has logic (heart five feeling; up arrow five up) and is thus opaque and requires learning (see Table 37.1). The more concrete a representation, the more recognizable it will be. However, representations of abstract meanings, such as emotions, will involve less-transparent images necessitating a longer learning curve (see Table 37.1)

One of the many challenges in the field of AAC is to provide novel, appropriate vocabulary for nonliterate children with CCN. Conventional speech output communication systems rely on literate helpers to store static vocabulary. This restricts the vocabulary of users as generation of new words require literacy skills. The ongoing BlissWord project (Andreasen, Waller, and Gregor 1998) is investigating ways in which users with physical and cognitive limitations can explore new vocabulary. Blissymbolics is a semantic-based natural written (graphic) language, similar to Chinese. Because of its generative characteristics (BlissWords are spelled using a sequence of one or more Bliss characters), predictive algorithms can be applied to Blissymbolics to assist users in the retrieval of words. BlissWords are sequenced beginning with a classifier (e.g., all emotions begin with a heart). As illustrated in Figure 37.2, selecting a shape from the Bliss keyboard, the interface produces a list of BlissWords which begin with classifiers using that shape. Frequency and word lists can be used further to refine the BlissWords that are displayed. Users do not need to be literate to explore language and vocabulary. It is envisaged that video clips and spoken explanation could further augment learning through exploration.

37.6.1 Use of Natural-Language Generation Techniques

Natural-language generation (NLG) is an area of research in natural language processing by computer. NLG systems have been harnessed to support language development for children with language disorders by providing opportunities to extend both their vocabulary and the type of conversation in which they can engage (Waller 2006). The STANDUP project (Waller et al. 2009) demonstrated the use of pun generating technology for children with CCN. A graphic-based interface

TABLE 37.1
Examples of Three Symbol Sets That Have Been Developed for Augmentative and Alternative Communication—for More Information See about These Symbol Systems

	PCS	Rebus	Blissymbols
House			
Happy			
Sad			
Big			
Small			
Fall			

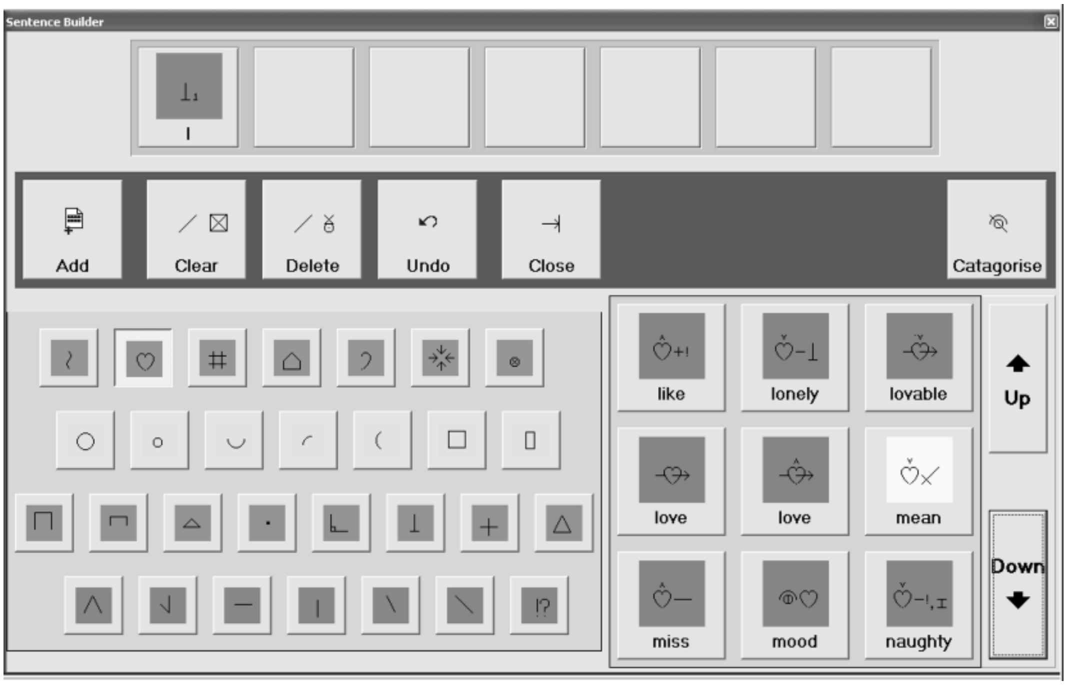


FIGURE 37.2 Screenshot showing Blissymbol keyboard on left and prediction list on right.

was designed to provide children with independent access to novel puns. An evaluation of the system with nine children with cerebral palsy showed that all the children were able to generate increasingly sophisticated puns. Language tests administered before and after a 4-week intervention program indicated an increase in the children's ability to categorize words into groups. Although no generalization can be made, it suggests that such systems can have an impact on underlying cognitive and language abilities.

Another project focusing on narrative illustrates how NLG can support the reporting of personal experience within conversations. Although it is relatively straight forward to provide script-based interactions and novel puns, conversational narratives emerge spontaneously within interactions. Early research into narrative has shown that nonspeaking individuals and their caregiverss had great difficulty in anticipating what personal experiences might make a good story (Waller et al. 1997). NLG systems which generate texts in English and other human languages from nonlinguistic input (Reiter and Dale 2000) can provide access to personal experience without the need for others to input information, especially when users are not literate. The "How was school today?" project (Reiter et al. 2009; Black et al. 2010) has produced a working prototype which uses sensor data to generate stories for conversations.

The prototype was developed in a special needs school which provided a constrained research environment. The system had access to three kinds of input as follows: (1) location, (2) interaction, and (3) voice messages. Location data was acquired automatically by using a radio-frequency identification (RFID) sensor which picked up RFID tags on doors while interaction data was provided by teachers who swiped cards through an RFID reader. Teachers could also record voice messages about interesting events during the day. The input data (filtered to reduce noise) was analyzed by a knowledge-based system which had access to a domain knowledge base and the child's daily timetable; this system identified a small number of reportable (i.e., interesting in the narrative sense) events. These were displayed (using graphical symbols) as a timeline; an editing interface allowed the child to delete events she did not wish to discuss, and also to add simple evaluations (annotations) such as "I liked it." Finally, a narration interface allowed the child to narrate the story. This interface was designed to support interactive story-telling, allowing the child to narrate events in any order and to add additional evaluations.

Results showed that this type of technology can both enhance interactive conversation and support storytelling skills and memory sequencing. The system could support simple but effective interaction and story-telling, enabling users to engage in extended conversations, rather than responding using single-word responses.

Reiter et al. (2009) have been exploring the use of NLG to produce conversational utterances in communication support systems designed for social interaction. The aim is to automate the production of readable or listenable text from abstract information held by the system. In the case of a communication system for a physically disabled nonspeaking person, the user would input only a small amount of information, which the system could then expand into a large number of utterances.

It is possible that using NLG might address some of the difficulties observed in prestorage systems. For instance, the generation component could theoretically produce a range of utterances and speech act types automatically from the same underlying data and adapt these somewhat to the interactional context. Using NLG would also have the benefit of offering control over the well-formedness of the output, an important consideration given the difficulty some AAC users have in achieving literacy. The fact that the system has an inherent awareness of the semantic content of the linguistic output, rather than simply being stored as canned text, is also a potential benefit. In other words, NLG might offer a level of automaticity and flexibility that traditional prestorage systems cannot offer, as well potentially as reducing the level of pre-authoring required from the user.

A growing line of inquiry in the NLG community is the generation of language from ontologies (Mellish and Sun 2005). An ontology is a logical and hierarchical model of the different concepts and the nature of relationships between concepts in a particular domain. These concepts and relationships can be mapped onto linguistic constructs to allow for the production of natural language descriptions of parts of the ontology. In a prototype AAC system, conversational topics that would be of interest in social conversation between users of the system and their co-conversationalists are modeled with ontology. The current categories of topic include travelling, listening to music, visiting the cinema, and attending concerts. Many categories are based on a simple event model that defines the basic characteristics common to all events, such as a time of occurrence. Concepts such as "person and place" are included, which are associated with events, to form a logical model of a particular event type. A separate knowledge base is created unique to each user which is linked to the original model. This is filled with data inputted by the user. In essence, what is being modeled is the Who, What, Where, When, and Why of different topics so that a set of utterances can be produced which describe them.

The knowledge base is then turned into useful conversational utterances through a template-driven utterance generation system. A large set of templates has been authored which convert the computer representation into natural language utterances. The templates are created as concrete syntax trees containing unspecified "slots" and parameters (figure). These syntax trees map out the syntactic structure of the template and are linked to a particular class in the ontology so that only appropriate templates are applied to each individual. Slots are used to add contextually relevant clauses to our utterances. For example, a template might contain a "time" slot, the contents of which are derived from the time of the event in question. For instance, the slot might be filled with "next Tuesday evening," "a month ago," or "this morning," depending on the context. Example parameters include the tense with which the utterance should be generated, and whether a pronoun or full noun phrase should be used to refer to the subject of the utterance.

Iterative evaluations of the system so far have concentrated on training the users in its operation, updating conversational material, and implementing changes based on the user feedback. In initial trials, it has been found possible to hold pleasing conversations lasting around 10 or 15 minutes with unfamiliar partners, with the aided communicator speaking at a rate of around 35–40 wpm. There also seems to be higher incidences of initiations on the part of the user, with them making good use of both the scripted NLG material, the quick fire phrases (including the contextual quick fire phrases), and their own prestored material. The topic progression feature is currently being underused but the users are responding well to training sessions on how best to use this to reduce their response time.

37.7 SUPPORT FOR APHASIA

Communication systems for nonspeaking people have been described, which in some way model the cognitive processes underlying communication. In the case of most physically disabled nonspeaking people, this is needed to speed up the communication process. However, in the case of speech problems that are caused by a stroke or other trauma (aphasia), the person trying to communicate will also have cognitive problems to deal with. Interestingly, the objections that conversation modeling might provide an active prompt to communication suggested a way of possibly helping people who might need such a prompt to initiate communication at all.

In a research project investigating the possibility of prompting people with Broca's aphasia in their communications, a predictive communication system was developed with a very simple interface (Waller et al. 1997). The system held personal sentences and stories, which were entered with the help of a caregiver. The user could then retrieve the prestored conversational items, with the system offering probable items based on previous use of the system. The interface was designed to be as simple as possible and to be usable by people who were unfamiliar with technology, had language difficulties, but had retained the ability to recognize familiar written topic words.

To access the sentences and stories, the user is led through a sequence of choices on the screen. First, they are offered a choice of conversational partners, a list of topics most likely to be appropriate for the chosen partner, and then a list of the four most common sentences for that partner and that topic. The user can choose to speak one of the sentences through a speech synthesizer, or have the system look for more suggestions. The sentences and topic categories are personal to each user, and the order in which topic words or sentences are presented depends on the past use of the system. Thus, the system is specific to users, both in the information content and in how it adapts to individual ways of communicating.

The system was evaluated with five adults with nonfluent aphasia, who were able to recognize but not produce familiar written sentences. There was little change in the underlying comprehension and expressive abilities of the participants while not using the system. When making use of the system, the results showed that some adults with nonfluent aphasia were able to initiate and retain control of the conversation to a greater extent when familiar sentences and narratives were predicted. In other words, users' existing/residual abilities (e.g., small vocabulary, pragmatic knowledge of conversation) were to a degree augmented by the computer functioning as a cognitive prosthesis. This project indicated that a communication system based on prompting could be of help to people with cognitive and communication difficulties.

37.8 SUPPORT FOR DEMENTIA

Dementia, which involves the loss of short term memory (STM) in elderly people, is a very serious problem for the person and for their families and caregivers. It can rule out most social activities and interactions, since these depend on a working STM for effective participation. This includes even the essential ability to communicate.

37.8.1 Support for Reminiscence

As well as being valuable with all older people, reminiscence is an important tool used to help elderly people who have dementia (Sheridan 1992; Feil 1993). This is because, while their short-term memories may be impaired, their long-term memories are often more or less intact (Rau 1993). The difficulty is accessing these long-term memories without the capability of keeping a conversation going, which depends on STM. Activities that do not require patients to maintain the structure of a conversation can help, for instance, looking at and commenting on a series of photographs, which can provide a framework for meaningful person-to-person interaction.

The tools used in such reminiscence work can also include videos, sound, music, and written materials. However, traditionally these are all in separate media, and it can be very time-consuming searching for a particular photo, sound, or film clip. Bringing all these media together into a digital multimedia scrapbook could mean easier access to content for more lively reminiscence sessions.

An investigation was undertaken to determine which aspects of multimedia would be most helpful for such a reminiscence experience, and the best way to present them. The intention of this project was to develop a system that could act as a conversation prosthesis, giving the user the support needed to carry out a satisfying conversation about the past. A number of prototype interfaces for a multimedia reminiscence experience were developed. These included text, photographs, videos, and songs from the past life of the city. The materials were collected with the assistance of the University, Dundee City archives, and the Dundee Heritage Project. The prototypes were demonstrated for people with dementia and their care staff at a day center run by Alzheimer Scotland Action on Dementia. The following issues were addressed in the following evaluation sessions:

1. Is it better for the display to use the metaphor of a real-life scrapbook or a standard computer screen? Six of the staff members preferred the book presentation, three preferred the screen, and two had no preference. Interestingly, this was almost a reversal of the preference shown by the people who had dementia, with the majority preferring the screen presentation. The preference shown for the screen presentation could be due to reduced cognitive ability. The book presentation is a metaphor that may not be interpreted suitably by the person with dementia.
2. How should the scrapbook material be organized—by subject or by media type? The majority of the staff evaluators preferred the arrangement by subject saying it was more logical, some were unsure; however, no one showed a preference for the arrangement by media. The clients with dementia reflected these findings. Despite preferring the arrangement being by subject, the majority of evaluators could see benefits from having access to both arrangements. It was concluded that for basic reminiscence sessions the arrangement by subject is preferable. But access to the arrangement by media should be an option, to make the software available for use in other ways.
3. How do the sounds, pictures, video, and music add to the reminiscence process, and what are their differential effects? Most of the videos and photographs and all the songs were able to spur conversations. However, it was found that with videos the clients were able only to identify strongly with them when they triggered off specific personal memories, whereas songs and photographs were more generally appreciated. Attention remained focused longest on the songs, which were particularly enjoyed when played repeatedly with everyone singing along. However, the staff felt that some individual clients had enjoyed the videos most.

One general finding was that the multimedia presentation as a whole produced a great deal of interest and motivation from the people with dementia. The staff was also very keen to see the idea developed further. In addition to its benefits for the person with dementia, participation in reminiscence activities has been shown to have a positive outcome for the caregivers who take part. Thus, help for reminiscence is not only a tool to stimulate interaction, but also a contributor to improved quality of life for the person with dementia and his or her family.

Recent work on using videos to present life histories for people with dementia has shown that new technologies, where sensitively and appropriately applied, can bring a substantial added impact to supportive and therapeutic activities for people with cognitive problems (Cohen 2000). We hope the reminiscence system developed will have the immediate application described above and will also serve as an exploration in developing a range of computer-based entertainment systems for people with dementia.

On the basis of this preliminary work, a communication support for people with dementia was developed (Alm et al. 2007). The system, called computer interactive reminiscence conversation aid (CIRCA), consists of a hypermedia structure with reminiscence material as content. It contains photographs, music, and video clips, all from the years 1930–1980. The system attempts to relieve the caregiver or relative of the job of continually supporting the person with dementia in a conversation. An obvious advantage of a computer-based system over using traditional materials is the bringing together of the various media into one easily accessible system. It was also speculated that such a hypermedia structure could in some way mimic the way memory is used in conversations, thus presenting the material in a way that seems natural in the context of a conversation.

An iterative design approach was used to develop the first prototype. Forty people with dementia were identified who expressed an interest in helping us to design and evaluate the system. Thirty relatives and caregiverss also agreed to take part as advisors and evaluators for the project. As a first step, the group commented on suitable content for the system, and ideas for themes which the system might include (such as national events, local industries, street life, and celebrations) were suggested by means of high-quality photographs. Using the photographs, discussions were held with people with dementia and their caregivers to help decide on the best choices for content for the system.

Based on feedback from the user population, the three themes chosen for the prototype were as follows: (1) recreation, (2) entertainment, and (3) local life. The system allowed the users to select and play video clips related to the theme selected. The video clips had a short duration because people with dementia may not be able to follow a long video clip because of their working memory (WM) problems. In addition, these clips were intended to act as conversation prompts and not be too immersive.

The system allowed the users to select and play music related to the theme selected. Music has proved be a powerful stimulant to long-term memory. Often people with dementia who have stopped speaking can be engaged by music and can proceed to sing entire songs word perfect with evident enjoyment. The difficulty in presenting musical prompts is the physical process of accessing a particular song and playing it without having to set up equipment and materials beforehand. Having touch-screen access to songs and music provided an instant way to produce a wide variety of musical prompts. Work by another research group on a touch screen "Picture Gramophone" with images as buttons to access music has established the usefulness of making familiar music instantly accessible for people with dementia (Topo et al. 2004). Rather than simply having a static screen while the music played, the system displayed the type of device that particular music would normally be coming from, with animated movement where appropriate. A record player, radio, or tape recorder was displayed depending on the theme selected. This representation of the music producing device acted as a conversation prompt in itself.

A series of evaluations were performed during the development of the prototype, and of the final prototype version. The prototype was evaluated with people across the range of dementia severity. People with mild to moderate dementia were able to make full use of the system. More severely affected people could not actually engage with changing the system display, but did show a marked reaction to any musical items which came up. The fact that music can reach even those with severe dementia is well known, but the difficulty of accessing music without setting up an organized session for it means that less music is used than caregivers would ideally like. The system was used both in one-to-one sessions and as the center of a group activity. CIRCA was used under conditions of close observation and also left at a residential facility for an extended period, with staff instructed to make whatever use of it they liked. Comparisons were made between sessions involving other enjoyable activities and sessions using the prototype.

One particular question was how using the prototype would compare with traditional reminiscence sessions. The original intention was to make the running of such sessions easier by means of multimedia accessed through a touch screen, and also to establish whether an interactive system had advantages over traditional reminiscence sessions. In a series of videotaped, transcribed, and coded evaluation runs, the system was compared directly with traditional reminiscence sessions. In a traditional session, the caregiver takes the responsibility for guiding the session, and at all points must compensate for the WM problems of the person with dementia. The system was designed to take over the role of supporting the cognitive abilities of the person with dementia, freeing the caregiver to take part in the session more naturally. Each person with dementia and their caregiver undertook a 20-minute reminiscence session: half using the prototype, and half using traditional reminiscence methods. As expected, the caregivers did most of the direct operation of the touch screen. However, both had their attention held by the displays on the screen and the caregiver was often prompted and directed by the person with dementia. With encouragement, a number of people with dementia also made direct use of the touch screen to make selections.

In a comparison with CIRCA with traditional reminiscence aids, it was found that the person with dementia was offered a choice of reminiscence subject/materials more often when using CIRCA ($U = 1.50$, $p < .001$). It was also found that the person with dementia chose reminiscence subjects/materials more often when prompted when using CIRCA ($U = 3.60$, $p < .001$). The traditional sessions were characteristically a series of one question from the caregiver followed by one response from the person with dementia. The CIRCA sessions were more of a conversation, with each person contributing an equal amount, and control of the direction of the conversation being shared. The caregivers asked more direct questions when using traditional reminiscence methods ($U = 5.00$, $p = .01$).

It was expected that providing reminiscence materials in a multimedia format on a large but not intimidating touch panel would have some advantages over traditional methods of delivering a reminiscence session. Reminiscence sessions are known to be valuable, but are not carried out as frequently as would be desirable, because of the preparation time necessary. Also they are normally performed as a group activity, because of considerations of the economies of staff time and availability. One-to-one reminiscence sessions could be very successful, but are not often possible in care settings. Demonstrating and evaluating the system elicited entirely positive reactions from care staff and relatives.

37.8.2 Making Music

Making music is a form of pleasurable communication that requires WM to perform. A project to develop a cognitive scaffolding to allow a person with dementia to compose and perform their own music has been developed, based on a simple but engaging touch-screen interface (Riley, Alm, and Newell 2009).

The system is called Express Play. The user is invited to select a mood (happy, sad, or angry), and then move to one of three screens where they can play either happy, sad, or angry music. The user is then prompted to "drag [their] finger around the screen to play music." When the screen is touched, two types of instant feedback are given simultaneously—audio and visual—and so users hear a chord playing while also seeing a trail of circles appear on the screen under their finger. These circular shapes not only provide instant and constant feedback, but also add interest to the screen. As the user's finger moves, so they begin to draw on the screen, leaving a trail of circles behind. This visual trail provides a continuous prompt that something happens when the screen is touched. This kind of prompting is particularly important for those with severe WM loss.

As the user touches different areas of the screen, chords play of different pitch and volume. Moving up the screen causes the pitch of the chord to become higher, whereas moving down the screen causes the pitch to become lower. The intention here was to provide an intuitive interface where a "higher" note was "higher" up the screen. Volume is dependent on where the user touches the screen on the horizontal axis. As the user moves to the right, the sound will become louder, and as they move to the left the sound will become quieter, in keeping with the normal stereotype of a volume control. As the intended user may have a hearing impairment, volume has been set so that it does not fall below a certain value. During use, if a finger is dragged from the top of the screen to the bottom of the screen, a passage of chords will be played in sequence (starting with a high pitched chord that will gradually get lower). The music produced should sound both harmonious and musical. By touching different parts of the screen, users will play chords of different pitch and volume and may be able to develop a tune, while also playing music that portrays the mood initially selected.

The system was developed iteratively, with the participation of people with dementia and caregivers throughout. A final formal evaluation was carried out to establish whether the system provided engagement, novel music (i.e., that it supported creativity), and music which was particular to each individual. In all three of these aspects, the system showed a positive effect (Riley 2010).

37.9 SOFTWARE TO SUPPORT DYSLEXIA

Both research and commercial software applications have been developed to alleviate some of the problems of dyslexia. Such software tends to provide options, for example, to have on-screen content read aloud and allow text adaptations such as color contrast, text size, and spacing between words and lines of text (e.g., Elkind 2009; *Web Design for Dyslexic Users* 2009). Such manipulations reduce problems with color contrasts and visual crowding and can reduce the amount of clutter on a page. In addition, there is strong evidence to suggest that the use of lexical and spelling aids can greatly assist with spelling problems exhibited by dyslexics (e.g., Newell and Booth 1991). However, merely highlighting an incorrect word and offering a replacement may not be enough, because one of the other problems that some dyslexics face is an inability to tell if a word looks right, thus they will have difficulty in selecting the appropriate corrections.

The importance of providing the ability for individual users to make changes for their own needs is highlighted in research with this population, showing great individual variability, for example, in the font options when using software that allows them to adapt pages to their own needs (Gregor and Newell 2000; Hanson et al. 2005). The following description of the development of one research tool for use by dyslexic readers is illustrative of this need for individual solutions.

The approach adopted in this research by Gregor and Newell (2000) was to offer dyslexic users a range of appropriate visual settings for the display of a word processor, together with the opportunity to configure easily the way in which text is displayed to them. The user can select, by experimentation, the settings that best suit them. These settings are then saved and later recalled each time that person uses the word processor. It will be seen that this approach affords the potential to make computer-based text significantly easier to read than printed text, as well as improving the usability of computer word-processing systems for a wide range of dyslexics.

The first stage of the research was to develop an experimental text reader. This prototype presented the user with an easily configurable interface, which allowed for a number of display variables to be altered. Initially, these were background, foreground and text colors, font size and style, and the spacing between paragraphs, lines, words, and characters. The interface was designed in such a way that it gave visual feedback on selections before they were confirmed and made minimal use of text instruction.

This was evaluated using 12 computer literate dyslexic students from higher education using think-aloud techniques, as well as questionnaires and interviews. At various development stages, the helpers were asked to try out the system with a view to seeing if it was possible for them to put together a display, which improved their abilities to read text from the screen. All the users were able to find a setting that was subjectively superior for them to standard black text on a white background with Times Roman 10- or 12-point type, but the screen layouts that were developed by the test subjects were extremely varied. This highlighted the individual nature of the disorder, and the diverse characteristics of any interface that would be appropriate for this group.

Each appeared to have a favorite color combination, although brown text on a green background was liked by all the testers. Subjects were in greatest agreement about the selection of a typeface: Sans Serif. Arial was rated the best by almost all the testers. All reported that increasing the spacing between the characters, words, and lines was beneficial. The most interesting point that arose during the testing, however, was the fact that at the beginning of the evaluation period the dyslexic subjects did not appear to be aware that altering these variables might be of any use.

A second prototype was then developed based on Word for Windows (Microsoft 1994, 1995) macros to provide the required configuration interfaces. This was based on the concept of an evolutionary system, rather than a fixed prototype. It was clear that there would be a substantial advantage in developing a dyslexic configuration, but this design decision raised an interesting deviation from the received wisdom of the desirability of WYSIWYG (What You See Is What You Get). In the case of a dyslexic user, what you see should be whatever you can read best, and print previewing facilities would have to be used to show how the layout will appear when printed.

This prototype provided a facility to enhance characters prone to reversals (e.g., b, d), by using color font type and size. This idea of coloring reversal characters provided very interesting and unanticipated results, which are described below. Fixation problems were tackled by reducing the page width, and a speech synthesizer that could read the text on the screen was included.

There were two distinct parts to the overall solution, a preference program and a reading/editing program. The first allowed users to experiment with the various parameters and the second made use of these preferences within a reading and editing environment. The preference program menu presents the user with various options and variables, together with a preview facility, to enable the user to experiment with, and finally store his or her data in a preferences file.

The fact that a unique user environment tailored to the need of each individual is provided means that the document is (deliberately) not WYSIWYG. A print option thus allows the user to print the document as it appeared with his or her preferred formatting applied to it, or as it would appear without any special formatting.

This second prototype was developed as an add-on module to Microsoft Word, and was evaluated by seven dyslexic users of 15–30 years old, in a similar fashion to that above. The users found the system easy and intuitive to use, reporting that each of the options had an effect on their abilities to read. The options that allowed the user to change the color scheme of the document appeared to be the most helpful, but font size and spacing, column width, and indications of reversals were also reported to assist reading by some or all of the users.

The reversals option provided the most interesting results of all. However, the reason for the improvement was not always that the reversal characters were clearly distinguished and easy to read. Instead, it was claimed that the sporadic coloring broke the text up and resulted in the user being less likely to get lost, for example, the system was reducing fixation problems rather than recognition problems.

As the testing progressed, the testers appeared to be surprised at times by the effect some of the changes had on their abilities to read the document. Comments included "I would never have thought of doing that," or "I don't think that will do me much good" before finding that a feature did indeed help.

The prototypes were developed from the perspective that the user population was diverse, and that the design process must accommodate potential changes in preferences over time. The fact that dyslexia is a very idiosyncratic disorder and findings that the users were often unaware of how easy it was to improve their reading potentials by changing visual aspects of the reading environment, illustrates how a standard user-centered design methodology is not appropriate for such user groups.

This development of this word processor to assist dyslexics is thus a particularly illuminating example of how the needs of people with a particular type of cognitive impairment can be effectively factored into the design and development process. This research has been described in some depth, as an example of a development and to assist the reader in appreciating the generic importance of this approach, which requires knowledge of the underlying syndrome, and a methodology, which encourages an innovative approach to user involvement.

37.10 COGNITIVE SUPPORT FOR ACTIVITIES OF DAILY LIVING

Moderate to severe dementia can affect the ability of a person to complete basic activities of daily living (ADL). At these later stages, people are usually assisted by a human caregiver, either a family member or a professional, who monitors the activity in question and guides or prompts the person when they are no longer able to make progress. This usually occurs in situations where the person cannot recall what activity they are attempting, cannot recognize important elements of their environment, or cannot perceive affordances of the environment that are critical for the task at hand. The dependence on a caregiver can lead to difficulties including loss of independence for the person in need, and increased burden on family members and other caregivers. These problems become acute for private ADLs such as using the washroom.

These acute problems with moderate or severe dementia can be aided with computerized cognitive assistive technologies (CATs), which are devices or systems that can help this population to complete ADLs more independently by monitoring them during a task, detecting when a problem occurs, and providing assistance automatically when necessary, and in extreme cases alerting a caregiver.

The School of Computing at the University of Dundee has been building a CAT for the ADL of hand washing in collaboration with groups at the University of Waterloo and the University of Toronto. The system uses a camera mounted above a sink to watch a person with dementia who is washing their hands. The video outputs from the camera are fed to a computer vision system that tracks the person's hands and the towel, and categorizes their actions into a set of behaviors, such as *using the soap*, and *turning on the water*. These behaviors are then passed to a monitoring system that maintains a *belief* about where the user currently is in the process of hand washing. The monitoring system uses a probabilistic and decision-theoretic model known as a partially observable Markov decision process (POMDP) that maintains a probability distribution (a *belief*) over the possible states the person could be in. It updates this belief over time as new observations of a person's behavior are detected by the computer vision system. For example, if the system believes that the user does not have soapy hands, an observation of the user's hands near the soap dispenser will make it more likely that the user's hands have become soapy.

The POMDP model then maps its *belief* about the person's progress to an *action* that it can take to help. For example, if the person has not made any progress in the past few time steps, then giving an audio prompt to the person may help them start again. The POMDP decides what *action* to take from a fixed set of possibilities based on a *reward* function. This reward function indicates which outcomes are desirable and which are not. For example, in hand washing, it is good if a person gets their hands clean, but not good if they require a lot of prompting and also not good if the human caregiver needs to be called in to assist very much. The POMDP decides on an action to take based on its long-term expected reward, so can effectively trade-off between short-term gains (e.g., calling a caregiver immediately, a costly action, to get a person to finish the task quickly and with certainty) versus long-term rewards (e.g., helping a person finish on their own, taking longer but preserving feelings of independence and reducing costly caregiver burden). This trade-off is made using decision-theoretic methods, which are justifiable based on the structure of the task and the reward function.

The actions that the POMDP decides upon are then relayed on the person trying to wash their hands as either audio prompts, audio-visual prompts (on a display screen mounted above the sink), or as a call for a human to assist.

This system was tested in a long-term care facility in Toronto, Canada. Six older adults participated in the study—five women, one man, and average age 86 years. Using the Mini-Mental State Examination, five of the subjects were classified as having moderate level dementia, with the one remaining classified as severe. Eight weeks of trials were carried out with each person washing his or her hands once per day. The trails used an A-B-A-B protocol, where in A phases the subjects were assisted by a human, whereas in the B phases, the system performed the assistance. Results showed that for some users, a 100% decrease in caregiver dependence was achieved. For others, the decrease was less, but overall we found a decrease of 25% in requirement for caregiver assistance when using the system.

The results highlight the user-specific nature of such systems, in that what works for one person may not work for another for a variety of reasons. The simplest such reason is the delivery of prompts. Whereas some users found the disembodied voice from a pair of speakers to be acceptable, others found it disconcerting, and would try to find out where the voice was coming from. Some users seemed to pay attention to the video prompts, others not. More details on this system can be found in Mihailidis et al. (2008); Hoey et al. (2010a).

Our work in developing CATs for persons with dementia has extended to other ADLs (mobility, toothbrushing, cooking, and rehabilitation [Kan et al. 2011]), to creative arts therapies (Blunsden et al. 2009; Hoey et al. 2010b; Mihailidis et al. 2010), and to other user groups such as children and adults with autism spectrum disorder. We are currently developing systems for use in the home as well as for use in long-term care facilities. Key to the uptake of such technology in users' homes is the ability for the end users and caregivers to customize the systems to suit their individual needs. This customization can range from the recording of individualized prompts, to the specification of new ADL for which a person requires assistance (Hoey et al. 2011).

37.11 BEYOND COGNITIVE PROSTHESES: THE POTENTIAL BENEFITS OF "BRAIN TRAINING"

There is an emerging interest in interactive systems that support the exercising of cognition with the aim of maintaining functionality and staving off the effects of advancing age. A broad range of research has indicated that age-related cognitive decline is modifiable rather than inevitable with many (positive and negative) factors being identified as influential (see Fillit et al. [2002] for a review). Positive factors such as, physical health, life-long learning, and a range of proactive interventions have been identified as beneficial; among these is the importance of generally keeping the mind active. This has promoted the current wide and increasing acceptance, particularly amongst older adults, of the "use-it-or-lose-it" adage. Various approaches addressing the need for older adults to "use-it" have emerged both in terms of traditional face-to-face activities and also using a variety of interactive systems. Although there are some limitations in the recent research in this area, there is certainly a strong suggestion that it can have benefits for many older adults (see Salthouse [2006] for a review). However, there is notable variation in the range and extent of the impact found and little consistency between the approaches adopted and thus more work will be required to more clearly identify who will benefit from what.

Beyond the distinction mentioned above, between human- and computer-based interventions, it is also worth noting the distinction between "contextualized" and "decontextualized" approaches, which are often referred to as cognitive rehabilitation and cognitive training, respectively (Clare et al. 2003). Simply stated, cognitive rehabilitation involves identifying and addressing individual day-to-day needs and goals, and providing coaching and practice for these in concrete terms, with related training for skills and strategies such as the use of appropriate memory aids. Whereas, cognitive training involves repeated practice with sets of relatively abstract tasks that reflect particular basic cognitive functions, such as memory, attention, or problem solving, which become "strengthened" and better able to deal with day-to-day challenges generally. Thus, cognitive rehabilitation tends to be targeted at those who have already suffered noticeable (and possibly problematic) declines in their cognitive ability; whereas, cognitive training could be of benefit to virtually anyone, but particularly those concerned about the onset of "normal" age-related decline.

It can be seen that these different approaches have differential scopes of impact and challenges to their implementation. The rehabilitation approach can be effective and produce measurable improvements in the targeted skills. However, the two caveats for this are that such improvements tend to be only for the targeted skills and not for (even very) closely related ones and there is also a tendency for the improvements to be relatively short lived (Wilson 1997). This appears to be mainly due to such approaches being based upon repeated drills and rote learning, which in most contexts will produce results, but will also (particularly for older adults) make it difficult to generalize and transfer that which is learned into analogous situations. Further, without repeated practice, the skill will become less effective or simply forgotten about. There also appears to be a further limitation, at least for the rehabilitation of memory problems in that the greater the need for memory enhancement, the less the apparent benefit (Verhaeghen et al. 1992).

In terms of cognitive training, the idea is to exercise and strengthen the component cognitive abilities involved in day-to-day tasks, rather than teach or train the task itself. As mentioned above, there is some evidence that this type of approach can be beneficial, but the wide variation in the types of intervention (and subject samples, etc.) makes it difficult to draw any overall conclusions. Although there are many commercial offerings claiming to be interactive "brain training," most are no more than interactive puzzle sets and few have had their effectiveness validated. However, some are based on the relevant cognitive and neuropsychological literature and support for their effectiveness is emerging in the literature (e.g. Smith et al. [2009]). Most of these rigorous cognitive training programs are provided via PC/Internet (rather than games consoles), and thus unfortunately exclude the many older adults who cannot afford or do not want such equipment. Alternatively, the emergence (and some might say, imposition) of digital television means that all households will soon need digital reception equipment, which has the potential to act as a platform for PC/Internet style services. At the time of writing, one of the authors is involved in an EC-funded project developing a neuropsychologically based cognitive training application compatible with digital television which will also take the first steps to validating any measurable impact on the cognitive status of a cohort of elderly users (Freeman et al. 2009). These training programs are tailored to individual users, usually on the basis of a comprehensive set of assessment tasks, which provide an ability profile, allowing a set of suitable training activities to be compiled which have their parameters set in accordance with the users' "strengths & weaknesses." The subsequent training series (usually of 2–3 months duration) is then set at a suitable level of challenge which taxes the user enough to produce beneficial results, but is also achievable enough to allow the users' motivation to continue to be maintained. To retain this optimum level of challenge over the longer term, the ongoing training follows an assessment-training-assessment cycle.

In general, interactive cognitive training is an emerging area with great potential but a relative paucity of research identifying the impact on people's day-to-day lives and providing suitable information for further development. However, regardless of valid measurable impacts obtained from specific approaches to training, it is clear that there is enormous scope to fend off the effects of age-related cognitive decline, at least to some extent. The main conclusion that can be drawn from the literature is that maintaining a lifestyle that incorporates (among other things) regular intellectual challenges is the best way to minimize the effects of advancing age. This suggests avoiding the sedentary lifestyle stereotypical of "the elderly" and replacing it with interesting and challenging activities; "Even in old age carrying out substantively complex tasks builds the capacity to deal with the intellectual challenges such complex environments provide" (Schooler and Mulatu 2001, p. 466).

For those who have already experienced significant cognitive decline, approaches more related to rehabilitation are required and thus more research is needed to build generalizable guidelines for the most effective approaches and more effort needs to go into providing useful support for those whose general cognitive function may not be amenable to improvement.

37.12 RESEARCH METHODOLOGIES

The research described above gives a flavor of successful approaches to developing human interfaces and software to support people with various types of cognitive impairment. Much of the methodology used in these developments, however, had to be developed *ab initio*. Traditional user-centered design does not have the flexibility for these user groups, and most research and development in the field of communication and information technology to support people with disabilities has, to date, concentrated on the development of special assistive systems and on accessibility features for younger, mainly physically or sensorially disabled people. Similarly, the human interfaces to most computer systems for general use have been designed, either deliberately or by default, for a typical, younger user (Newell and Cairns 1993; Newell 1995; Newell and Gregor 1997). Knowledge from these fields does not necessarily transfer comfortably to the challenges encompassed in universal design (Beirmann 1997; Sleeman 1998; Stephanidis 2001) and, in particular, the widely varying and often declining abilities associated with the range of cognitive impairments.

This section addresses the particular issues for the design process which accompany cognitive impairment and suggests a paradigm and methodology to support the process of designing software that is as near to the universal accessibility ideal as is possible, derived from the approach to specific projects described above.

Software systems, which are aimed at the mainstream (rather than being of a prosthetic nature) need to address the wide variation in the types and severity of cognitive impairment between individuals. This demand is further complicated by the fact that as people grow older their abilities change. This process of change includes a decline over time in the cognitive, physical, and sensory functions, and each of these will decline at different rates relative to one another for each individual. This pattern of capabilities varies widely between individuals, and as people grow older, the variability between people increases. In addition, any given individual's capabilities vary in the short-term due, for example, to temporary decrease in, or loss of, function due to a variety of causes, such as the effects of drugs, illness, blood sugar levels, and state of arousal.

This broad range and variability of change presents a fundamental problem for the designers of computing systems, whether they be generic systems for use by all ages, or specific systems to compensate for loss of function. Systems tend to be developed for a typical user, and either by design or by default, this user tends to be young, fit, male and crucially, has abilities that are static over time. These abilities are assumed broadly similar for everybody. Not only is this view wrong, in that it does not take account of the wide diversity of abilities among the wider population of users, but it also ignores the fact that for individuals, these abilities are dynamic over time.

Current software design typically produces an artifact which is static and which has no, or very limited, means of adapting to the changing needs of users as their abilities change. Even the user-centered paradigm (e.g., ISO 13407 1999; Nielsen 1993; Preece 1994; Shneiderman 1992) looks typically at issues such as representative user groups, without regard for the fact that the user is not a static entity. Thus, it is important not only to be aware of the diverse characteristics of people with cognitive dysfunction, but also the dynamic aspects of their abilities.

It is clear that people with cognitive impairments, whatever their cause, can have very different characteristics to most human-interface and software designers. It is also clear that in these circumstances user-centered design principles need to be used if appropriate technology is to be developed for this user group (Gregor and Newell 1999). However, these methodologies have been developed for user groups with relatively homogenous characteristics. People with dementia, for example, are a diverse group and even small subsets of this group tend to have a greater diversity of functionality than is found in groups of able young people.

An additional complication is that there can be serious ethical issues related to the use of such people as participants in the software development process. Some of these are medically related, but also include, for example, the ability to obtain informed consent. It is thus suggested that the standard methodology of user-centered design is not appropriate for designing for the inclusion of this user group. The importance of research and development taking into account the full diversity of the potential user population, including cognitive diversity was addressed by Newell in his keynote address to InterCHI '93, where the concept of "Ordinary and Extraordinary Human Computer Interaction" was developed (Newell and Cairns 1993; Newell and Gregor 1997).

Market share is clearly an important consideration, and this has been given impetus, not only by demographic trends, but also by recent legislation in the United States and other countries, on accessibility of computer systems for people with disabilities. In terms of the workplace, both the Americans with Disabilities Act and the United Kingdom Disability Discrimination Act put significant requirements on employers to ensure that people with disabilities are able to be employed within companies and to provide appropriate technology so that such employees had full access to the equipment and information necessary for their employment. Increasingly, there is political pressure to increase this access, and more requirements for improved access by disabled people are being enshrined in legislation. However, access does not mean only that people with wheelchairs can maneuver around buildings; it also means that there needs to be provision for people with cognitive (and sensory and other physical) impairments to be able to operate computers and other equipment essential to the workplace.

An important additional factor in the value for money equation is that design that takes into account the needs of those with slight or moderate cognitive dysfunction can produce better design for everyone. An example where this has not occurred is illustrated by the problems that the majority of users have had with video tape recorders. If the designers had considered those with cognitive impairments within their user group, they may have been able to design more usable systems. Another example is an e-mail system specifically designed to be simple to use by older people with reduced cognitive functioning, which was found to be preferred by executives to the standard e-mail system that they were used to.

Some people are impaired from birth, but some may become temporarily or permanently disabled by accident or illness (suddenly or more slowly), or even by normal functioning within their employment. This is particularly noticeable in cognitive functioning. Short-term changes in cognitive ability occur with everyone. These can be caused by fatigue, noise levels, blood sugar fluctuations, lapses in concentration, stress, or a combination of such factors, and can produce significant changes over minutes, hours, or days. In addition, alcohol and drugs can also induce serious changes in cognitive functioning, which is recognized in driving legislation, but not in terms of how easy it is to use computer-based systems.

Most people at one time or another, will exhibit cognitive functional characteristics that are significantly outside the normal range. Although neither they, nor their peers, would consider these people disabled, their abilities to operate standard equipment may well be significantly reduced.

The questions that designers need to consider include the following:

- Does the equipment that I provide comply with the legislation concerning use by employees who may be cognitively disabled?
- To what extent do I need to take into account the needs of employees who are not considered disabled, but have significant temporary or permanent cognitive dysfunction?
- Should I make specific accommodation for the known reductions in cognitive abilities which occur as employees get older (e.g., less requirement for STM, or the need to learn new operating procedures)?
- What are the specific obligations designers and employers have to provide systems that can be operated by employees whose cognitive ability has been reduced due to the stress, noise, or other characteristics of the workplace?

The argument is that it would be very unusual for anyone to go through their working life without at some stage, or many stages, being significantly cognitively disabled. If equipment designers considered this, it is probable that the effectiveness and efficiency of the work force could be maintained at a higher level than would be the case if the design of the equipment were based on an idealistic model of the characteristics of users and their work environments.

37.12.1 Disabling Environment

In addition to the user having characteristics that can be considered disabled, it is also possible for them to be disabled by the environments within which they have to operate. Newell and Cairns (1993) made the point that the human-machine interaction problems of an able bodied (ordinary) person operating in an high workload, high stress or otherwise extreme (e.g., extraordinary) environment has very close parallels with a disabled (extraordinary) person, operating in an ordinary situation (e.g., an office).

High workloads and the stress levels to which this can lead often reduce the cognitive performance of the human operator. For example, a very noisy environment cannot only create a similar situation to hearing or speech impairment, but can also lead to reduced cognitive performance. The stress level in the dealing room of financial houses can be very high and is often accompanied by high noise levels. A significant advance may be made if the software that was to be used in these houses was to be designed on the assumption that the users would be hearing impaired and have a relatively low cognitive performance. It is interesting to speculate as to whether such systems would produce higher productivity, better decision making, and less stress on the operators. Other examples of extreme environments in which people have to operate are the battlefield, under water, or out in space. The stress and fatigue caused by working within such environments means that a soldier's performance may be similar to that which could be achieved by a very disabled person operating in a more normal environment. It is not always clear that the equipment such people need to operate has been designed with this view of the user.

It is very important to describe the users of technology in terms of their functional abilities related to technology rather than generic definitions of either medical conditions, or primarily medical descriptions of their disabilities. Unfortunately, most statistical data is presented as generic and medically categorizations of disability. Gill and Shipley (1999), however, defined disabled user groups in terms of their functional abilities, with specific emphasis on the use of the telephone. They estimated that within the European Union, which has a population of 385 million, there were 9 million people with cognitive impairment that could lead to problems using the telephone. These figures do not take into account multiple impairments, and the authors pointed out that, in the elderly population in particular, there may be a tendency toward cognitive, hearing, vision, and mobility impairments being present to a varying extent and these may interact when considering the use of technological systems. It is this multiple minor reduction in function (often together with a major disability), which means that the challenges to technological support for older people have significantly different characteristics to that of younger disabled people and to the nondisabled, nonelderly population.

There has been some movement in mainstream research and development in technology, both in academia and industry, away from a technology led focus to a more user led approach, and this has led to the development of user-centered design principles and practices in many industries. In addition, a number of initiatives have been launched to promote a consideration of people with disabilities within the user group in mainstream product development teams with titles including "Universal Design," "Design for All," "Accessible Design," and "Inclusive Design." (http://www.design.ncsu .edu/cud/ud/ud.html. http://www.stakes.fi/include. http://www.trace.wisc .edu. http://www.w3.org/WAI) The "Design for All/Universal Design" movement has been very valuable in raising the profile of disabled users of products, and has laid down some important principles. However, their approach has tended not to place too much significance on cognitive impairment, and, particularly if this is included as a factor in the design process, then it becomes more difficult to use traditional user-centered design approaches.

Newell and Gregor (2000) suggested that a new design approach should be developed, which would be based on the already accepted user-centered design methodology. There are some important distinctions between traditional user-centered design with able-bodied users, and the approach needed when the user group either contains, or is exclusively made up of, people with cognitive dysfunction. These include the following:

- Much greater variety of user characteristics and functionality.
- The difficulty in finding and recruiting representative users.

- Situations where design for all is certainly not appropriate (e.g., where the task requires a high level of cognitive ability).
- The need to specify exactly the characteristics and functionality of the user group.
- Conflicts of interest between user groups, including temporarily able bodied.
- Tailored, personalizable, and adaptive interfaces.
- Provision for accessibility using additional components (hardware and software).

The balance in the design process also needs to shift from a focus on user needs to one on the users themselves. There will be additional problems when considering people with cognitive dysfunction, which will include the following:

- The lack of a truly representative user group.
- That a different attitude of mind of the designer is required.
- Ethical issues (Alm 1994; Balandin and Raghavendra 1999).
- It may be difficult to get informed consent from some users.
- Difficulties of communication with users.
- The users may not be able to (sufficiently) articulate their thoughts, or even may be incompetent in a legal sense.

Thus, there can be particularly difficult ethical problems when involving users with cognitive impairments in the design process. In addition, it is often necessary to involve clinicians when such users are involved, so some of the user-centered design actually focuses on professional advice about the user, rather than direct involvement of the user. Even with these problems, however, it is possible to include users with cognitive dysfunction sensitively in the design process.

37.12.2 Inclusion of Users with Disabilities within Research Groups

In Dundee users with disabilities have a substantial involvement in the research, and they have made a significant contribution both to the research and to the commercial products that have grown from this research. Users are involved in two major ways as follows:

- As disabled consultants on the research team, where they act essentially as test pilots for prototype systems.
- By the traditional user-centered design methodology of having user panels, formal case studies, and individual users who assess and evaluate the prototypes produced as part of the research.

The contribution made by clinicians is also vital to the research, and these are full members of the research team. Dundee's Applied Computing Department is also one of the few computing departments that have had speech therapists, nurses, special education teachers, linguists, and psychologists (both clinical and cognitive) as full-time researchers.

37.12.3 User Sensitive Inclusive Design

Some significant changes must be introduced to the user-centered design paradigm if users with disabilities are to be included, and this is particularly important if the users have cognitive impairment. To ensure that these differences are fully recognized by the field, the title "User Sensitive Inclusive Design" has been suggested. The use of the term inclusive rather than universal reflects the view that inclusiveness is a more achievable, and, in many situations, appropriate goal than universal design or design for all. Sensitive replaces centered to underline the extra levels of difficulty involved when the range of functionality and characteristics of the user groups can be so great that it is impossible neither to produce a small representative sample of the user group in a meaningful way, nor often to design a product that truly is accessible by all potential users.

37.12.4 Design for Dynamic Diversity

In addition to the aspects of user sensitive inclusive design described above, it is necessary to make designers fully aware of the range of diversity which can be expected with cognitively impaired people, and also the changing nature of the cognitive functioning of people. Thus, Gregor and Newell (2000) suggested that this be drawn particularly to the attention of designers by introducing the concept of "Designing for Dynamic Diversity." This process, described above, entails recognition that people's abilities are diverse at any given age and that as they grow older this diversity grows dynamically; it also involves a recognition that any given individual's abilities will vary according to factors such as mood, fatigue, blood sugar levels, and so on. Only by taking on board, the factors associated with Designing for Dynamic Diversity will software design produce artifacts which are not static and which have no, or very limited, means of adapting to the changing needs of users as their abilities change.

As has been seen above, metaphors and processes in use at present are limited in meeting the needs of this design paradigm or addressing the dynamic nature of diversity. New processes and practices are needed to address the design issues; awareness raising among the design, economic and political communities has to start; and research is needed to find methods to pin down this moving target.

37.12.4.1 Story-Telling Metaphor

In addition, researchers need to consider how best to disseminate the concepts behind universal usability and the results of user sensitive inclusive research. User sensitive inclusive design needs to be an attitude of mind rather than simply the mechanistic application of design for all guidelines. This offers a further challenge to the community. The dangers of using such studies to produce more extensive guidelines has been referred to above, but it is important that the results of user sensitive inclusive design are made available to other designers and researchers. However, it is too early to lay down principles and practices that must be followed by designers, and it may even be impossible to do this for some of the contexts and environments in which designers work. Thus, it is suggested that we follow a story-telling approach, in which information about accessibility issues, and design methods which focus on accessibility is presented in narrative form, with particular examples to illustrate generic principles. This is, in some sense, an extension of the single case-study methodology. This methodology could provide very useful insights to designers in a form that they will find easy to assimilate and act. Thus, this will assist in their educations and will help them to design more accessible products, and better products for everyone.

37.12.5 Use of Theater

As an extension of the story-telling metaphor, the research group in the School of Computing at Dundee University has investigated the use of dramatic techniques and theater as a way of addressing the challenges of user sensitive design. In particular, they have investigated the use of professional theater for both awareness raising with designers and also for requirements gathering with older adults (Newell et al. 2006a). They have commissioned a number of live performances and professional narrative videos to illustrate the output of long-term research into the challenges older people find with new technologies. These have been produced as an educational tool for human interface engineers, software designers, managers, and procurement executives. They are designed to provoke and facilitate discussion with both developers and potential uses of technology about the needs and wants of older people. Morgan and Newell (2007) describe the methodology of using professional theater in research of this nature.

37.12.6 Theater for Awareness Raising

In the UTOPIA (Usable Technology for Older People: Inclusive and Appropriate) project (Dickinson et al. 2002), they worked in collaboration with the Foxtrot Theater Company (Perth, Scotland) to use theater to encourage interaction between (older) users of technology and designers. The outcome was the "UTOPIA Trilogy," a series of short-video plays addressing problems that older people have in using technology (Carmichael et al. 2005). The films were developed to be amusing and entertaining dramatizations of some of the issues the researchers had encountered during the project. These films were based on real events, conversations, and observations, and they were the amalgamation of many and are intended to convey older people's experiences with technology and the situations they encounter. They are amusing and entertaining. These videos were evaluated with a variety of audiences including academics, practitioners, software engineers, relevant groups of undergraduates, and older people. This established that the videos provided a useful channel for communication between users of technology and designers, and changed the perceptions of both students and more mature designers of IT systems and products about older people's requirements. Subsequently in collaboration with MMTraining (http://www.MMTraining.org) whose artistic director was the artistic director of the Foxtrot Theater Company (http://www.foxtrot-theatre.org.uk) and Soundsmove (http://soundsmove.com), a professional video production company. They have produced a range of videos and live theater events. The videos "Relative Confusion" and "Relatively PC" (which can be viewed at http://www.computing.dundee.ac.uk/projects/iden) address the issues older people can have with digital television and the use of the Internet, respectively (Newell 2009). In an amusing way, these videos illustrated many of these challenges older people can have with the technology including the following: users' ability to learn and their memory for new control methods, the effects of poor eyesight and manual dexterity, the interaction of poor eyesight and memory, loss of control due to complex interaction techniques, visual distractions intergenerational differences, the consequences of jargon, operational anxiety, and the effects of stressreasons for technophobia, and the challenges provided by technical language and metaphors. Although older users are represented in these videos, the lessons illustrated apply to many other groups of naïve and cognitive impaired users.

37.12.7 Theater for Requirements Gathering

A similar technique has also been used in the requirements gathering phase for IT systems. Within a project developing systems designed to monitor older people in case of falls at home (Marquis-Faulkes et al. 2003, 2005), a series of short films were produced that illustrated how such systems worked and gave examples of ways in which they may operate inappropriately. In keeping with our methodology (Morgan and Newell 2007), these were narratives rather than documentaries and they used conflict and humor to present questions to the audience. Groups of older people were shown the videos, which were used to facilitate discussion on the characteristics of the system that the users required. Although it was not appropriate within this project to make comparisons with traditional focus groups, the use of theater did produce very lively discussion and the authors believe that it was unlikely that some of the conclusions would have been arrived at without this type of presentation.

Rice, Newell, and Morgan (2007) describe the use of live theater for requirements gathering for a project investigating potential applications of interactive television for older people. This technique was particularly useful for the conceptual stages of the design process. The researchers were investigating a number of different systems (including a video phone and a "memory box"). The use of theater with its "suspension of disbelief" and use of props meant that the use these systems could be illustrated in a clear and meaningful way in real contexts at a predevelopment stage. Morgan et al. (2008b) have used live theater in their investigations of the required characteristics of supportive environments for older people. Focus groups of older people, facilitated by live theater produced very useful data. As a demonstration of this technique, they mounted a live theater event at CHI 2008. A play showing a couple of older people living within a "supportive home of the future" was followed by a session in which the audience was encouraged to question the actors who stayed in role. This produced a very lively discussion session. The use of theater has proved to be a very useful way to encourage audience discussion at both this and other international conferences (Newell and Morgan 2006, 2008a,b,c; Morgan et al. 2008a).

37.12.8 Use of Professional Actors to Simulate Users with Dementia

The research described above has focused on older people without any major cognitive dysfunction. In his research into cognitive support for daily living (see Section 37.10), Hoey (2010c) has conducted a pilot project where he used actors to simulate people with dementia. He used actors, who had been well briefed in the characteristics of people with dementia to provide training data for an adaptive prompting system. There would have been significant ethical issues in the use of people with dementia at this stage of the project, and the data gathering would have taken very much longer. This use of actors is being continued in a joint project with the University of Toronto.

37.12.9 Conclusions on the Use of Theater

This research has shown that the use of theater can be a very powerful method of encouraging dialogue between various professional groups particularly in a clinical environment, for keeping a focus for discussions, and also for providing a channel for communication between users of technology and designers (Carmichael, Newell, and Morgan 2007). The researchers view is that the success of this approach was in large part due to the plays being narrative based rather than having a pedagogic style. That is, they illustrated the issues involved within interesting story lines, with all the characteristics of a good narrative—humor, tension, human interest, and antagonists and protagonists. In addition, the quality of the production, having been produced by theater professionals, played a major part in the success of the venture.

Newell et al. (2006b) discussed the various ways in which actors and theater can play a part in the design process for human–computer interfaces. This could provide a particularly valuable methodology for the design process when the target users have cognitive impairment and thus may not be appropriate for including within standard user-centered design methodologies. For example, the use of actors removes any ethical issues which may arise if researchers were to use "real people," especially those with cognitive dysfunction, for early experiments with very novel technology and/or to illustrate challenges that novel technologies may provide.

37.13 USE OF ART-SCHOOL-TRAINED DESIGNERS

37.13.1 Esthetics

Researchers working with people with dementia have identified a heightened artistic appreciation and emotional response (Pullin 2009a, p. 83—citing Orpwood et al. 2005). This would indicate that the esthetic design of objects for people with dementia or other cognitive dysfunction is just as important as—perhaps even more important than in mainstream design. However, there is a tendency for "rehabilitation technology" to exhibit symptoms of a lack of esthetic considerations in the design process. Many such products show evidence that the teams that design them do not engage emotionally with the users groups and assume that older and disabled people lack any esthetic sense, and, unlike other user groups, are motivated entirely by the functionality of products.

There is an interesting contrast between hearing aids and eyeglasses in this context. Hearing aids have always been considered to be a medical product and have been designed to be as invisible as possible. However, eyeglasses moved from being a medical product that, in the 1930s, were described by government as "needing to be adequate," to a fashion accessory by the 1980s (Lewis 2001). Ironically, unlike eyeglasses, most hearing aid users would benefit greatly from advertising the fact that they were hearing impaired in the sense that this would encourage conversational partners to speak more clearly. In a further contrast, a range of very fashionable wheelchairs existed in the nineteenth century, but wheelchairs became much more obviously a medical product for most of the twentieth century. This changed in the 1970s due to pressure from Veterans Association of North America, and a much more varied and exciting range of wheelchairs became available in the latter part of the 1990s (Woods and Watson 2004). Thus, although some branches of assistive technology have developed as fashionable devices, this is the exception rather than the rule. There can be an assumption that the additional constraints involved in considering older and disabled people mean the abandonment of novel and beautiful concepts. If these attitudinal constraints are over emphasized, the design team will be focused exclusively on the ergonomic and technical aspects of the product, which can lead to products that patronize and further stigmatize the very people they are designed to help.

This lack of sensitivity to the cultural needs of users of assistive technology, and the lack of esthetic considerations and empathy between the designers and the customers could well be a factor in the very high level of abandonment of assistive technology of products. Hocking (1999) reports that in the United States, 56% assistive technology is quickly abandoned, and 15% are never used.

The culture of problem solving, particularly prevalent in rehabilitation engineering circles, can often see fashion as the antithesis of good design. Thus, creative designers are less likely to be part of an assistive technology development team. To ensure that software and hardware products are esthetically pleasing, it is important that the design team does include art-school trained industrial and interaction designers as well as medical engineers, human–computer interaction experts, clinical professionals, caregivers, and the people with cognitive impairment themselves (Pullin 2009a, p. 83). In addition art-school trained designers can contribute a range of design techniques which are not usually found in engineering design, and these can be particularly valuable when designing for cognitively impaired users. These include experience prototyping and critical design.

37.13.2 Experience Prototyping

There are a range of ethical issues when using participatory design techniques when the user population has cognitive impairments. An earlier section has suggested that actors could act as substitutes in some situations, but there will be points in the design process, however, when people with actual cognitive impairment will provide an invaluable input, as in some cases their reactions may be counter intuitive, even to their own caregivers. For example Orpwood et al. (2005) describes the development of an audio prompting system where experiments with people with dementia suggested voices for the prompts that neither the caregivers nor the researchers would have predicted. Early and unreliable prototypes, however, can be unsettling, even upsetting to users with cognitive impairments. In these cases, there is an argument for employing "experience prototyping," which often employ hidden human intervention (Pullin 2009a, p. 146). Such prototypes may be more faithful to the users' eventual experience, even at the expense of serving as technical proof of concept prototypes at the same time.

37.13.3 Critical Design

Critical design is the developments of artifacts which are intended to be provocative rather than an attempt at a solution to the design challenge (Dunne 2006). This provides a way of exploring the design space in a playful and open ended way but with a serious purpose. Critical design prototypes may challenge existing cultural, technological, and economic values. They may well address taboo issues or unspoken aspects of the context of use of devices, and may well instantiate these challenges by uncomfortable images and dark humor. They are designed to be used to encourage different insights and perspectives on the issues raised, to inspire new paths, and challenge assumptions on which the product was built. Although at first sight critical design may seem wasteful and self-indulgent, Pullin (2009a, pp. 111–133) gives examples of critical design prototypes and describes a range of issues related to design for people with disabilities which could benefit from such an approach.

37.13.4 Advantages of "Creative Design"

Art schools focus on the overall experience of using an artifact and attach significant value to exploring and feeling, to simplicity and to provocation, identify and expression, and the complex web of sensory and contextual interactions that determine whether a product succeeds or fails. Pullin (2009a) argues that the inclusion of such processes creates an important focus on people's engagement with the experience of using the product and the emotions this generates—these being important complements to the accessibility and usability of the product. It can also encourage an awareness of the important issues of self-confidence and security, and the users feeling of comfort when using a product, all of which can be influenced by the details of the design rather than just its functionality. People with disabilities should not be denied the esthetic pleasure of using devices. This is an important characteristic of many mainstream products, but assistive technology is often designed with esthetics being considered as an afterthought—the final "cosmetic treatment" of a product. As with accessibility, however, such considerations are much more effective and less expensive to provide if considered from the beginning as an integral part of the total design process. Pullin (2009a, p. 173) gives an example of how such characteristics were introduced in a simple communication aid for nonspeaking people, and how these contributed to its final effectiveness. He argues that there is almost inevitably a creative tension between such designers and engineering designers, but that these need to be embraced because anything less is a receipt for mediocrity, which would be more insulting to the potential users of the device than controversy in the design stages. Pullin (2009a) also includes discussions with a range of designers who were not familiar with assistive technology and shows how the contributions of such people could influence the design of assistive technology.

37.13.5 Appealing to Envy

The vast majority of people use both electronic and non-electronic cognitive prostheses—such as diaries, memo pads, alarm clocks, personal digital organizers. The popularity of individual products is due not only to their functionality but also to their ease of use and their design. Some are clearly a fashion accessory as well as a cognitive prosthesis. If these considerations were more obviously considered within the design of the prosthesis for cognitively impaired uses, then they would have a greater chance of being successful. An additional advantage of this approach could well be that such cognitive prostheses find a mainstream market. Newell and Cairns (1993) describes how this has occurred for a range of devices intended initially for use by disabled people, including the typewriter, the cassette tape recorder, and the ball point pen. More recent examples include word prediction and disambiguation as now available in the vast majority of mobile telephones. There is no reason why well-designed cognitive prostheses should not follow this path.

37.14 CONCLUSION

A range of information technology systems have been developed that successfully support people with cognitive impairment. In order to realize the full potential of this technology, however, a great deal more work is required. New knowledge about how cognition works is required and both specialist and mainstream designers need to be aware of the implications of this knowledge.

Although it is not necessary for software developers and human interface designers to be fully versed in all aspects of cognition, it is important for them to have some background knowledge of the area. Because of the wide range of skills and knowledge needed to understand the problems faced by people with cognitive impairment, research work in this field should be multidisciplinary, including psychologists, members of the health and therapeutic professions, and human interface, interaction, and creative designers. It is also vital to involve potential users of the technology as partners at all stages of the research and development of systems and products.

The development of the concept of, and a methodology for, user sensitive inclusive design, design for dynamic diversity, and development of story-telling methods for communicating results will facilitate researchers in this specialized field and will provide mainstream engineers with an effective and efficient way of including people with disabilities within the potential user groups for their projects. If both of these can be achieved, it will go some way towards providing appropriate technological support for people with cognitive impairment. As Christopher Frayling (2003) said: “Let’s bring the users in and let’s bring delight (back) into (everyday) products.” One measure of success might be that of non-cognitively impaired people being envious of the user of a particular device—wishing that they had one.

REFERENCES

Alm, N. 1994. Ethical issues in AAC research. In *Methodological Issues in Research in Augmentative and Alternative Communication: Proceedings of the Third ISAAC Research Symposium*, ed. J. Brodin and E. B. Ajessibm, 98–104. Sweden: University Press.

Alm, N., J. L. Arnott, and A. F. Newell. 1992. Prediction and conversational momentum in an augmentative communication system. *Commun ACM* 35(5):46–57.

Alm, N., R. Dye, G. Gowans, J. Campbell, A. Astell, and M. Ellis. 2007. A communication support system for older people with dementia. *IEEE Comput* 40(5):35–41.

Alm, N., A. Morrison, and J. L. Arnott. 1995. A communication system based on scripts, plans and goals for enabling non-speaking people to conduct telephone conversations. In *Proceedings of the IEEE Conference on Systems, Man & Cybernetics*, 2408–12. Vancouver, Canada.

Andreasen, P. N., A. Waller, and P. Gregor. 1998. BlissWord—full access to blissymbols for all users. In *Proceedings of the 8th Biennial Conference of ISAAC*, 167–8. Dublin, Ireland: ISAAC.

Arch, A. 2008. WAI-AGE Literature Review and Analysis: Observations and Conclusions [Editor’s DRAFT—21 August 2008] http://www.w3.org/WAI/WAI-AGE/conclude.html.

Bäckman, L., B. J. Small, Å. Wahlin, and M. Larsson. 2000. Cognitive functioning in very old age. In *The Handbook of Aging and Cognition*, ed. F. I. M. Craik and T. A. Salthouse, 499–558. New Jersey, NJ: Lawrence Erlbaum Associates.

Balandin, S., and P. Raghavendra. 1999. Challenging oppression: Augmented communicators’ involvement in AAC research. In *Augmentative and Alternative Communication, New Directions in Research and Practice*, ed. F. T. Loncke, J. Clibbens, H. H. Arvidson, and L. L. Lloyd, 262–77. London: Whurr.

Beirmann, A. W. 1997. *More than Screen Deep—Towards an Every-Citizen Interface to the National Information Infrastructure. Computer Science and Telecommunications Board, National Research Council*. Washington, DC: National Academy Press.

Beukelman, D. R., and P. Mirenda. 1998. *Augmentative and Alternative Communication Management of Severe Communication Disorders in Children and Adults*, 2nd ed., Baltimore, MD: Brookes.

Black, R., J. Reddington, E. Reiter, N. Tintarev, and A. Waller. 2010. Using NLG and sensors to support personal narrative for children with complex communication needs. In *First Workshop on Speech and Language Processing for Assistive Technologies (SLPAT), Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Los Angeles. Association of Comp Linguistics.

Blunsden, S., B. Richards, T. Bartindale, D. Dan Jackson, P. Patrick Olivier, J. Boger, A. Mihailidis, and J. Hoey. 2009. Design and prototype of a device to engage cognitively disabled older adults in visual artwork. In *Proceedings of the ACM 2nd International Conference on PErvasive Technologies Related to Assistive Environments*. Corfu, Greece: ACM.

British Dyslexia Association. 2006. http://www.bdaydyslexia.org.uk/facq.html#q1 (accessed 2010).

Carmichael, A. R. 1999. *Style Guide for the Design of Interactive Television Services for Elderly Viewers*. Winchester: Independent Television Commission.

Carmichael, A., A. F. Newell, and M. Morgan. 2007. The efficacy of narrative video for raising awareness in ICT designers about older users’ requirements. *Interact Comput* 19:587–96.

Carmichael, A., A. Newell, M. Morgan, A. Dickinson, and O. Mival. 2005. Using theatre and film to represent user requirements. In *Proceedings INCLUDE ’05*. London: Royal College of Art. ISBN (CD rom) 1-905000-10-3.

Charness, N., and E. A. Bosman. 1994. Age-related changes in perceptual and psychomotor performance: Implications for engineering design. *Exp Aging Res* 20(1):45–61.

Chin, J., and W. Fu. 2010. Interactive effects of age and interface differences on search strategies and performance. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems (Atlanta, Georgia, USA, April 10–15, 2010). CHI ’10*, 403–12. New York: ACM.

Clare, L., R. T. Woods, E. D. Moniz Cook, M. Orrell, and A. Spector. 2003. Cognitive rehabilitation and cognitive training for early-stage Alzheimer’s disease and vascular dementia. The Cochrane Database of Systematic Reviews: CD003260. Cochrane Library, Issue 4.

Cohen, G. 2000. Two new intergenerational interventions for Alzheimer’s disease patients and their families. *Am J Alzheimers Dis* 15(3):137–42.

Coyne, K. P., and J. Nielsen. 2002. *Web Usability for Senior Citizens—Design Guidelines Based on Usability Studies with People Age 65 and Older*. 126. Nielsen Norman Group. An overview is available at http://www.useit.com/alertbox/seniors.html. (accessed 2010).

Critchley, M. 1964. *Developmental Dyslexia*. London: Heinemann.

Czaja, S. J., N. Charness, A. D. Fisk, C. Hertzog, S. N. Nair, and W. Rogers. 2006. Factors predicting the use of technology: Findings from the center for research and education on aging and technology enhancement (CREATE). *Psychol Aging* 21(2):333–52.

Czaja, S. J., J. Sharit, M. A. Hernandez, S. N. Nair, and D. Loewenstein. 2010. Variability among older adults in Internet health information-seeking performance. *Gerontechnology* 9(1):46–55.

Detterman, D. K., L. T. Gabriel, and J. M. Ruthsatz. 2000. Intelligence and mental retardation. In *Handbook of Intelligence*, ed. R. J. Sternberg, 141–58. Cambridge, MA: Cambridge University Press.

Dickinson, A., R. Eisma, A. Syme, and P. Gregor. 2002. UTOPIA: Usable technology for older people: Inclusive and appropriate. *Proc. BCS HCI 2002* 38–9.

Dunne, A. 2006. *Electronic Products, Aesthetic Experience, and Critical Design*. Boston, MA: MIT Press.

Dyslexia Research Institute. 2010. http://www.dyslexia-add.org/ (accessed July 7, 2010).

Elkind, J. 1998. Computer reading machines for poor readers. *Perspectives* 24(2):4–6.

Fairweather, P. G. 2008. How older and younger adults differ in their approach to problem solving on a complex website. In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility (Halifax, Nova Scotia, Canada, October 13–15, 2008). Assets '08*, 67–72. New York: ACM.

Feil, N. 1993. *The Validation Breakthrough*. Maryland: Health Professions Press.

Fillit, H. M., R. N. Butler, A. W. O'Connell, M. S. Albert, J. E. Birren, C. W. Cotman, W. T. Greenough et al. 2002. Achieving and maintaining cognitive vitality with aging. *Mayo Clin Proc* 77:681–96.

Frayling, C. 2003. Interviewed on Desert Island Disks, BBC Radio 4, Nov 2nd.

Freeman, J., A. Miotto, J. Lessiter, A. De Gloria, F. Bellotti, M. Mangarone, N. Hofshi et al. 2009. Defining a framework to support cognitive training for older people via interactive digital television. IBC2009. U.K.: Independent Broadcasting Company.

Freudenthal, A. 1999. *The Design of Home Appliances for Young and Old Consumers*. Delft: Delft University Press.

Gill, J., and T. Shipley. 1999. *Telephones, What Features Do People Need*. London: Royal National Institute for the Deaf.

Gould, S. J. 1997. *The Mismeasure of Man*, 2nd ed., Harmondsworth: Penguin.

Gregor, P., and A. F. Newell. 1999. The application of computing technology to interpersonal communication at the university of Dundee's department of applied computing. *Technol Disabil* 10:107–13.

Gregor, P., A. F. Newell, and M. Zajicek. 2002. Designing for dynamic diversity—interfaces older people. In *Proceedings of the 5th International ACM SIGCAPH Conference on Assistive Technologies (ASSETS '02)*, 151–6. New York: ACM.

Hanson, V. L. 2009. Age and web access: the next generation. In *Proceedings of the 2009 International Cross-Disciplinary Conference on Web Accessibility (W4A) (Madrid, Spain, April 20–21, 2009). W4A '09*, 7–15. New York: ACM.

Hanson, V. L., J. Brezin, S. Crayne, S. Keates, R. Kjeldsen, J. T. Richards, C. Swart, and S. Trewin. 2005. Improving Web accessibility through an enhanced open-source browser. *IBM Syst J* 44(3):573–88.

Hanson, V. L., J. T. Richards, S. Harper, and S. Trewin. 2009. Web accessibility. In *The Universal Access Handbook*, ed. C. Stephanidis. Boca Raton, FL: CRC Press.

Hocking, C. 1999. Function or feelings: factors in abandonment of assistive devices. *Technol Disabil* 11:3–11.

Hoey, J., P. Poupart, A. von Bertoldi, T. Craig, C. Boutilier, and A. Mihailidis. 2010a. Automated handwashing assistance for persons with dementia using video and a partially observable markov decision process. *Comput Vis Image Underst* 114(5).

Hoey, J., K. Zutis, V. Leuty, and A. Mihailidis. 2010b. A Tool to promote prolonged engagement in art therapy: Design and development from art therapist requirements. In *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*, 211–18. Orlando, FL.

Hoey, J., J. N. Boger, K. Fenton, T. Craig, and A. Mihailidis. 2010c. Using actors to develop technologies for older adults with dementia: A pilot study. *Gerontechnology* 9(4).

Hoey, J., T. Ploetz, D. Jackson, P. Olivier, A. Monk, and C. Pham. 2011. Rapid specification and automated generation of prompting systems to assist people with dementia. Pervasive and Mobile Computing 7(3).

Holt, B. J., and R. W. Morrell. 2002. Guidelines for web site design for older adults: The ultimate influence of cognitive factors. In *Older Adults, Health Information, and the World Wide Web*, ed. R. W. Morrell, 109–29. Hillsdale, NJ: Erlbaum.

Kan, P., R. Huq, J. Hoey, R. Goetschalckx, and A. Mihailidis. 2011. The development of an adaptive upper-limb stroke rehabilitation robotic system. *Journal of NeuroEngineering and Rehabilitation* 8(33).

Kaluger, G., and C. L. Kolson. 1987. *Reading and Learning Disabilities*, 2nd ed., Columbus, OH: Bell & Howell Company.

Kaufman, A. S. 2000. Tests of intelligence. In *Handbook of Intelligence*, ed. R. J. Sternberg, 445–76. Cambridge, MA: Cambridge University Press.

Kurniawan, S. H. 2009. Age-related differences in the interface design process. In *The Universal Access Handbook*, ed. C. Stephanidis, Chap 8, 1–12. Boca Raton, FL: CRC Press.

Lewis, J. 2001. *Vision for Britain: The NHS, the Optical Industry and Spectacle Design 1946–1986*. MA dissertation Royal College of Art, London, U.K.

Lohman, D. F. 2000. Complex information processing and intelligence. In *Handbook of Intelligence*, ed. R. J. Sternberg, 285–340. Cambridge, MA: Cambridge University Press.

Mandler, J. M. 1984. *Stories, Scripts and Scenes: Aspects of Schema-Theory*. Mahwah, NJ: Lawrence Erlbaum Associates.

Mandler, J. M., and R. E. Parker. 1976. Memory for descriptive and spatial information in complex pictures. *J Exp Psychol Hum Learn Mem* 2:38–48.

Marquis-Faulkes, F., S. J. McKenna, P. Gregor, and A. F. Newell. 2003. Scenario-based drama as a tool for investigating user requirements with application to home monitoring for elderly-people. In *Human-Centred Computing: Vol. 3. Cognitive, Social and Ergonomic Aspects*, ed. D. Harris, V. Duffy, M. Smith, and C. Stephanidis, 512–6. Mahwah, NJ: Lawrence Erlbaum.

Marquis-Faulkes, F., S. J. McKenna, P. Gregor, and A. F. Newell. 2005. Gathering the requirements for a fall monitor using drama and video with older people. *Technol Disabil* 17(4):227–36.

McKinlay, A. 1991. Using a social approach in the development of a communication aid to achieve perceived communicative com-petence. In *Proceedings of the 14th Annual Conference of the Rehabilitation Engineers Society of North America*, ed. J. Presperin, 204–6. Washington, DC: The RESNA Press.

Mellish, C., and X. Sun. 2005. Natural language directed inference in the presentation of ontologies. In *Proceedings of the 10th European Workshop on Natural Language Generation*, 118–24.

Microsoft. 1994. *Word Developers Kit*. Redmond, WA: Microsoft Press.

Microsoft. 1995. *Word for Windows 95*. Redmond, WA: Microsoft Corporation.

Mihailidis, A., J. Boger, M. Candido, and J. Hoey. 2008. The COACH prompting system to assist older adults with dementia through handwashing: An efficacy study. *BMC Geriatr* 8(28). doi:10.1186/1471-2318-8-28.

Mihailidis, A., S. Blunsden, J. N. Boger, B. Richards, K. Zutis, L. Young, and J. Hoey. 2010. Towards the development of a technology for art therapy and dementia: Definition of needs and design constraints. *The Arts in Psychotherapy* 37(4).

Morgan, M., V. Hanson, C. Martin, J. Hughes, and A. F. Newell. 2008b. "Accessibility challenges—a game show investigating the accessibility of computer systems for disabled people." In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*. Florence, Italy: ACM.

Morgan, M., M. McGee-Lennon, N. Hine, J. Arnott, J. Martin, M. J. Clark, and M. Wolters. 2008b. "Requirements gathering with diverse user groups and stakeholders." In *CHI 2008 Proceedings*, 2597–600. Florence, Italy: ACM.

Morgan, M., and A. Newell. 2007. Interface between two disciples, the development of theatre as a research tool. In *Lecture Notes in Computer Science 4550*, 184–93. Springer.

Morrell, R. W., S. R. Dailey, C. Feldman, C. G. Mayhorn, and K. V. Echt. 2002. *Older Adults and Information Technology: A Compendium of Scientific Research and Web Site Accessibility Guidelines*. Bethesda, MD: National Institute on Aging. For a summary, see also "Making your web site senior friendly: A checklist." http://www.nlm.nih.gov/pubs/checklist.pdf (accessed 2010).

National Center for Learning Disabilities. 2007. http://www.ncld.org/content/view/447/391/ (accessed December 20, 2007).

Newell, A. F. 1995. Extra-ordinary human computer operation. In *Extra-Ordinary Human-Computer Interaction*, ed. A. D. N. Edwards, 3–18. Cambridge, MA: Cambridge University Press.

Newell, A. F. 2009. *"Educational videos: Examining the issues older people have in using modern technology," Interfaces 80 Autumn*. 18–9.

Newell, A. F., J. L. Arnott, L. Booth, and W. Beattie. 1992. Effect of the PAL word prediction system on the quality and quantity of text generation. *Augment Altern Commun* 8:304–11.

Newell, A. F., and L. Booth. 1991. The use of lexical and spelling aids with dyslexics. In *Computers & Literacy Skills*, ed. C. Singleton, 35–44. Hull: University of Hull.

Newell, A. F., L. Booth, and W. Beattie. 1991. Predictive text entry with PAL and children with learning difficulties. *Br J Educ Technol* 22:23–40.

Newell, A. F., and A. Y. Cairns. 1993. Designing for extra-ordinary users. *Ergon Des Hum Factors Ergon Soc* 10–16.

Newell, A. F., A. Carmichael, M. Morgan, and A. Dickinson. 2006a. The use of theatre in requirements gathering and usability studies. *Interact Comput* 18:996–1011.

Newell, A. F., and P. Gregor. 1997. Human computer interfaces for people with disabilities. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. K. Landauer, and P. Prabhu, 813–24. Amsterdam: Elsevier.

Newell, A. F., and P. Gregor. 2000. User sensitive inclusive design—in search of a new paradigm. In *Proceedings of the ACM Conference on Universal Usability*, 39–44. Washington, DC.

Newell, A. F., and M. Morgan. 2006. "The use of theatre in HCI research." In *"Engage" 20th Annual BCS HCI Conference.* UK: University of London. ACM.

Newell, A. F., and M. Morgan. 2008a. "An intelligent future"—interactive theatrical presentations addressing the issues of ambient intelligence and older people. In *Collaboration with MM Training. Vision in Action—Accessibility to Next Generation Networks, COST219ter Conference*. Brussels: COST 219.

Newell, A. F., and M. Morgan. 2008b. "An interactive theatre presentation: Intelligible Transport information systems—a user perspective" in collaboration with MM Training. In *"Access and the City" Conference, Dublin City Council, the Centre for Excellence in Universal Design, and the National Disability Authority*. Dublin, Ireland: Centre for Excellence in Universal Design.

Newell, A. F., and M. Morgan. 2008c. "An intelligent future"—interactive theatrical presentations addressing the issues of ambient intelligence and older people. In *Collaboration with MM Training, eInclusion Ministerial Conference*, Vienna.

Newell, A. F., M. E. Morgan, P. Gregor, and A. Carmichael. 2006b. CHI 2006. In *Experience Report in CHI 2006 Extended Abstracts on Human Factors in Computing Systems Montreal*, 111–7. Quebec, Canada: ACM.

Nielsen, J. 1993. *Usability Engineering*. London: Academic Press.

Ogozalec, V. Z. 1997. A comparison of the use of text and multimedia interfaces to provide information to the elderly. In *Proceedings of CHI '97*, 65–71. Atlanta, Georgia, New York: ACM Press.

Orpwood, R., C. Gibbs, T. Adlam, R. Faulkner, and D. Neegagawatte. 2005. *The Design of Smart Houses for People with Dementia: User Interface Aspects*, Vol. 4, 156–64. Universal Access in the Information Society.

Park, D. C. 1992. Applied cognitive aging research. In *The Handbook of Aging and Cognition*, ed. F. I. M. Craik and T. A. Salthouse, 449–93. Hillsdale, NJ: Erlbaum.

Pirkl, J. J. 1994. *Transgenerational Design, Products for an Aging Population*. New York: Van Nostrand Reinhold.

Preece, J. 1994. *A Guide to Usability—Human Factors in Computing*. London: Addison Wesley & Open University.

Pullin, G. 2009a. *Design Meets Disability*. Cambridge, MA: The MIT Press.

Pullin, G. 2009b. An introduction to universal design. *Dwell* 10(4):102–8.

Rabbitt, P. M. A. 1993. Does it all go together when it goes? The nineteenth bartlett memorial lecture. *Q J Exp Psychol* 46A(3):385–434.

Rau, M. T. 1993. *Coping with Communication Challenges in Alzheimer's Disease*. California: Singular Publishing Group Inc.

Reiter, E., and R. Dale. 2000. *Building Natural-Language Generation Systems*. Cambridge, U.K.: Cambridge University Press.

Reiter, E., R. Turner, N. Alm, R. Black, M. Dempster, and A. Waller. 2009. Using NLG to help language-impaired users tell stories and participate in social dialogues. In *Proceedings of the 12th European Workshop on Natural Language Generation*, 1–8. Athens, Greece: Association for Computer Linguistics.

Rice, M., A. F. Newell, and M. Morgan. 2007. Forum theatre as a requirement gathering methodology in the design of a home telecommunication system for older adults. *Behav Inf Technol* 26(4):323–32.

Riley, P. 2010. *Unpublished PhD Thesis*. Dundee, Scotland, UK: University of Dundee.

Riley, P. J., N. Alm, and A. F. Newell. 2009. An interactive tool to support musical creativity in people with dementia. *J Comput Hum Behav* 25:599–608.

Salthouse, T. 1985. *A Theory of Cognitive Aging*. Amsterdam: North Holland.

Salthouse, T. A. 1991. *Theoretical Perspectives on Cognitive Aging*. Mahwah, NJ: Lawrence Erlbaum Associates.

Salthouse, T. 2006. Mental exercise and mental aging: evaluating the validity of the 'use it or lose it' hypothesis. *Perspect Psychol Sci* 1:68–87.

Sayago, S., L. Camacho, and J. Blat. 2009. Evaluation of techniques defined in WCAG 2.0 with older people. In *Proceedings of the 2009 International Cross-Disciplinary Conference on Web Accessibililty (W4a) (Madrid, Spain, April 20–21, 2009). W4A '09*, 79–82. New York: ACM.

Schank, R., and R. Abelson. 1977. *Scripts, Plans, Goals, and Understanding*. Mahwah, NJ: Lawrence Erlbaum Associates.

Schooler, C., and M. Mulatu. 2001. The reciprocal effects of leisure time activities and intellectual functioning in older people: A longitudinal analysis. *Psychol Aging* 16(3):466–82.

Shaywitz, S. E., B. A. Shaywitz, K. R. Pugh, R. K. Fulbright, R. T. Constable, W. E. Mencl, D. P. Shankweiler et al. 1998. Functional disruption in the organization of the brain for reading in dyslexia. *Proc Natl Acad Sci U S A* 95(5):2636–41.

Sheridan, C. 1992. *Failure-Free Activities for the Alzheimer's Patient*. London: Macmillan Press.

Shneiderman, B. 1992. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Reading, MA: Addison-Wesley.

Sleeman, K. D. 1998. Disability's new paradigm, implications for assistive technology and universal design. In *Improving the Quality of Life for the European Citizen: Vol. 4 Assistive Technology Research Series*, ed. I. Placencia Porrero and E. Ballabio, xx–xxiv. Amsterdam: IOS Press.

Smith, G., P. Housen, K. Yaffe, R. Ruff, R. Kennison, H. Mahncke, and E. Zelinski. 2009. A cognitive training program designed based on principles of brain plasticity: Results from the improvement in memory with plasticity-based adaptive cognitive training study. *J Am Geriatr Soc* 57(4):594–603.

Stephanidis, C. ed. 2001. *User Interfaces for All*. Mahwah, NJ: Lawrence Erlbaum Associates.

Sternberg, R. J. 2000. The concept of intelligence. In *Handbook of Intelligence*, ed. R. J. Sternberg, 3–15. Cambridge, MA: Cambridge University Press.

Todman, J., L. Elder, and N. Alm. 1995. Evaluation of the content of computer-aided conversations. *Augment Altern Commun* 11(4):229–34.

Topo, P., O. Mäki, K. Saarikalle, N. Clarke, E. Begley, S. Cahill, J. Arenlind et al. 2004. Assessment of a music-based multimedia program for people with dementia. *Dementia* 3:331–50.

Vanderheiden, P. B., P. W. Demasco, K. F. McCoy, and C. A. Pennington. 1996. A preliminary study into schema-based access and organization of re-usable text in AAC. In *Proceedings of the RESNA '96 Conference*, 59–61. Salt Lake City, Utah, Arlington, VA: RESMA Press.

Verhaeghen, P., A. Marcoen, and L. Goossens. 1992. Improving memory performance in the aged through mnemonic training: A meta-analytic study. *Psychol Aging* 7:242–51.

Waller, A. 1992. *Providing Narratives in an Augmentative Communication System*. Unpublished doctoral dissertation. Dundee, Scotland, UK: University of Dundee.

Waller, A. 2006. "Communication access to conversational narrative." *Top Lang Disord* 26(3):221–39.

Waller, A., R. Black, D. O'Mara, H. Pain, G. Ritchie, and R. Manurung. 2009. "Evaluating the STANDUP pun generating software with children with cerebral palsy." *ACM Trans Access Comput* 1(3):27.

Waller, A., F. Dennis, J. Brodie, and A. Y. Cairns. 1997. "Evaluating the use of TalksBac, a predictive communication device for non-fluent aphasic adults." *Int J Lang Commun Disord* 33:45–70.

Web design for dyslexic users. 2009. http://www.dyslexia.com/qaweb.htm (accessed July, 2010).

Willows, D. M., R. S. Kruk, and E. Corcos. 1993. *Visual Processes in Reading and Reading Disabilities*. Mahwah, NJ: Lawrence Erlbaum Associates.

Wilson, A. 2003. Communicating with pictures and symbols. *CALL Centre Publications (Augmentative and Alternative Communication in Practice: Scotland)*. Edinburgh: University of Edinburgh. ISBN 1-898042-25-X.

Wilson, B. A. 1997. Cognitive Rehabilitation: How it is and how it might be. *J Int Neuropsychol Soc* 3:487–96. Cambridge University Press.

Woods, B., and N. Watson. 2004. A glimpse at the social and technological history of wheelchairs. *Int J Ther Rehabil* 11(9):407–10.

Zajicek, M. 2003. Patterns for encapsulating speech interface design solutions for older adults. In *Proceedings of the 2003 Conference on Universal Usability (Vancouver, British Columbia, Canada, November 10–11, 2003). CUU '03*, 54–60. New York: ACM.

Zajicek, M. 2007. Web 2.0: Hype or happiness? In *Proceedings W4A '07*, 35–9. New York: ACM.

38 Perceptual Impairments
New Advancements Promoting Technological Access

Julie A. Jacko, V. Kathlene Leonard, Molly A. McClellan, and Ingrid U. Scott

CONTENTS

38.1 INTRODUCTION

The introduction of early computers facilitated new ways for individuals with visual impairment to access information electronically: magnified, in Braille, or aurally via the conversion of digital information. However, the introduction of graphical user interfaces (GUIs) that present digital information via visual metaphors and icons contributed to the digital divide, which can hamper the productivity of this population. In many cases, even access to documents and forms can be a difficult to impossible task when available through the direct manipulation paradigm (Fortuin and Omata 2004). The exclusive reliance of GUIs on the visual interaction paradigm therefore threatens to limit accessibility for anyone whose visual channel is compromised (Dix et al. 1998). This chapter provides readers with (1) an introduction to the visual sensory channel; (2) a review of research approaches, models, and theories that are relevant to human–computer interaction (HCI) and visual impairment; and (3) a discussion of forms of visual dysfunction in the research, design, and evaluation of human–computer systems.

The interaction strategies and related interaction barriers for individuals with visual impairments in the past 15 years has received growing attention in an attempt to inform judicious, inclusive design for accessible information technologies (e.g., Assistive Technologies for Independent Aging: Opportunities and Challenges 2004; Arditi 2002; Brewster, Wright, and Edwards 1994; Craven 2003; Fortuin and Omata 2004; Fraser and Gutwin 2000; Gaver 1989; Jacko 1999; Jacko, Barnard, et al. 2004; Jacko et al. 2002; Jacko et al. 2000; Jacko et al. 2005; Jacko et al. 1999a,b; Jacko and Sears 1998). Visual impairments (that do not lead to blindness) can create barriers to distinguishing fine details of iconic screen targets and to tracking the highly dynamic nature of the pointer used to manipulate these icons (Fraser and Gutwin

2000). This is largely attributed to difficulty in manipulating objects with the pointer due to reduced visual acuity and visual field.

In terms of the visual sensory channel, it is known that user behavior is strongly influenced by the nature and amount of residual vision the user experiences in combination with computer interface characteristics. As an extreme example, a blind user without any functional vision will use fundamentally different coping skills to navigate an interface as compared to an individual with clouded vision due to cataracts (Jacko and Sears 1998). Harper, Goble, and Stevens (2001) emphasized that the differences in orientation, navigation, travel, and mobility of visually impaired versus sighted individuals should be considered in the design of technology because there are differences in the mental map and cognitive processes that occur across the spectrum of visual abilities.

The impetus for this chapter is two parts. First, the number of individuals who report low vision is anticipated to rise sharply with the aging baby boomers (who are, on average, living longer) as they experience age-related changes to their functional vision (e.g., reduced visual acuity, presbyopia, contrast sensitivity, color sensitivity, depth perception, and glare sensitivity). They are increasingly predisposed to acquire ocular diseases associated with older age (e.g., macular degeneration, diabetic retinopathy, glaucoma, and cataracts; for a review, see Orr [1998]; Schieber [1994]). Secondly, the digital divide imposed on the population with visual impairments has been measured in terms of technology access and unemployment (Gerber and Kirchner 2001). Looking beyond the United States, as the need for information increases globally so does the diversity of the people requiring access. As a result, a potentially large number of users may be disadvantaged with respect to gaining access to a variety of types of information without adequate accommodations.

The framework for the structure of this chapter results from an HCI approach first introduced by Jacko and Vitense (2001), and further clarified by Jacko, Vitense, and Scott (2003). Initial work in this research area included a comprehensive review of the literature to facilitate the development of a categorization scheme to account for categories of impairment. From this literature review, five major categories emerged: (1) hearing impairments, (2) mental impairments, (3) physical impairments, (4) speech impairments, and (5) visual impairments. Figure 38.1 illustrates that each of the five, overarching categories is composed of a collection of clinical diagnoses unique to that category (depicted in Figure 38.1 by $A_1,\ldots, A_n$, $B_1,\ldots, B_n$, $C_1,\ldots, C_n$, $D_1,\ldots, D_n$ and $E_1,\ldots, E_n$). Each diagnosis, in turn, influences certain functional capabilities that are critical to the access of information technologies (depicted in Figure 38.1 by $Y_1,\ldots, Y_n$). A subset of these functional capabilities can be directly linked to specific classes of technologies (shown at the bottom of the diagram).

While this framework is applicable to the five identified categories of impairment, its discussion and demonstrated utility in the scope of this chapter will address only visual

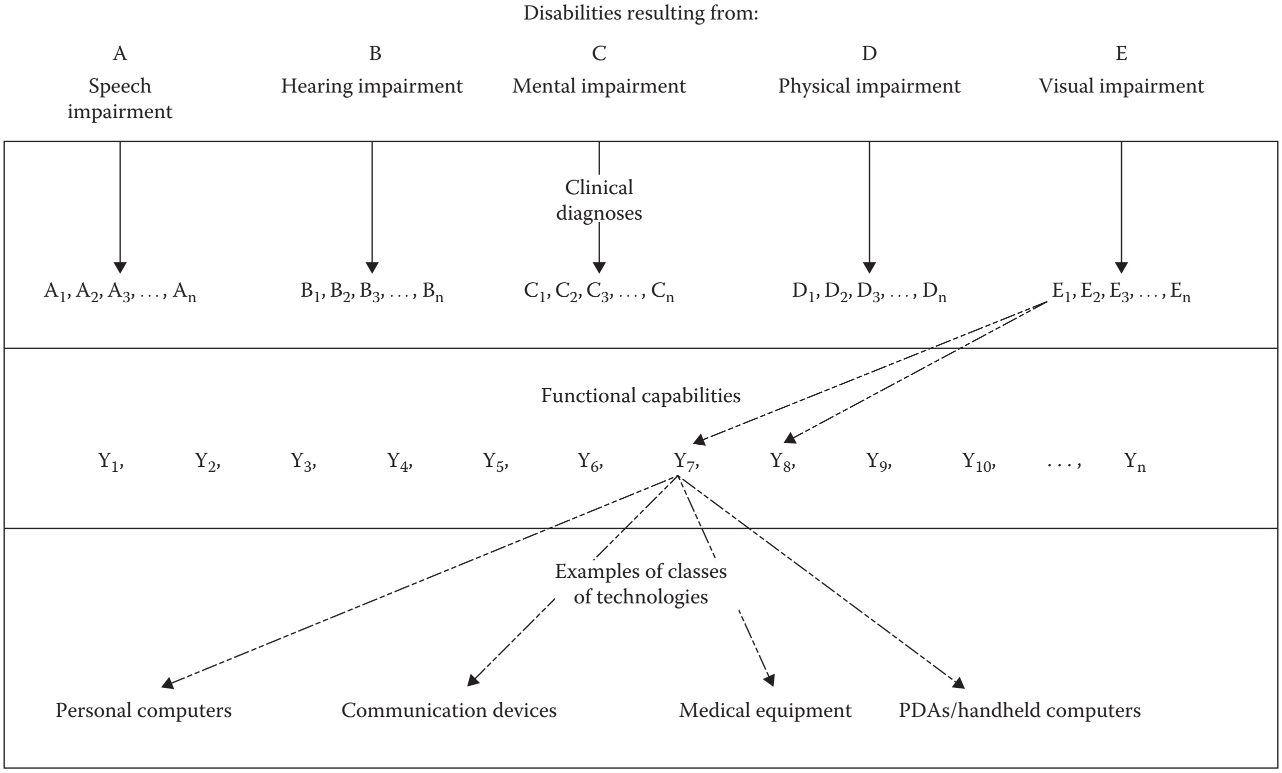


FIGURE 38.1 Framework for the integration of clinical diagnoses, functional capabilities, and access to classes of technologies. (Adapted from Jacko, J. A., H. S. Vitense, and I. U. Scott. 2003. Perceptual impairments and computing technologies. In *Human-Computer Interaction Handbook*, ed. J. A. Jacko and A. Sears, 504–22. Mahwah, NJ: Lawrence Erlbaum Associates.)

impairment. Auditory/speech, cognitive impairments and physical (motor) impairments are covered in this handbook in Chapters 35, 37, and 40. Consider, for example, a person who has been diagnosed with a specific type of visual impairment represented as E_1 in Figure 38.1. This visual impairment results in measurable decrements to certain aspects of this person's functional visual capabilities, Y_7 and Y_8. Observe from Figure 38.1 that the decrement to the functional capability represented Y_8 does not impede a person's access to any of the technology classes depicted in Figure 38.1. In contrast, the functional visual capability represented by Y_7 in Figure 38.1, which impedes this person's ability to successfully access information using all four examples of information technology. From this conceptual representation, it is apparent that much more knowledge is needed for researchers to possess an accurate depiction of the empirical relationships that exist between diagnoses, functional capabilities, and access to specific classes of technologies. More specifically, emerging from this conceptual framework are several key research areas in need of investigation as follows:

1. It is critical to establish empirical links between clinical diagnoses and sets of functional capabilities.
2. It is necessary to define the set of functional capabilities required to access information technologies.
3. It is essential to establish empirical bases for the influence of specific functional capabilities on access to specific classes of technologies.

Following this framework, this chapter introduces readers to the clinical definition of visual impairments and diagnoses, in which visual function is first addressed, followed by a discussion of the leading causes of vision loss in the United States and beyond. Section 38.2.4 also provides a discussion of specific visual functions. Then, the chapter highlights recent advancements in HCI research, which effectively link the three major areas of the framework: clinical diagnosis, functional impairment, and interaction across a variety of classes of technology and impairments. This research is done in pursuit of HCI solutions for the visually impaired including perceptual interfaces, multimedia interfaces, multimodal interfaces, and adaptive interfaces. Finally, examples of technological advancements for this population are presented. While these technological advancements are discussed briefly in this chapter within the context of visual impairments, it should be noted that additional information on HCI and perceptual-motor interactions, hearing and speech impairments, and motor impairments can be found in Chapters 35, 37, and 40.

38.2 VISUAL FUNCTION

38.2.1 Definitions

When considering the impact of visual loss on an individual's ability to use a computer effectively, it is first necessary to understand various dimensions of visual performance. Many different terms have been used to refer to abnormalities in visual function, including disorder, impairment, disability, and handicap (Colenbrander 1977). Although often used as synonyms, there are distinct differences. For instance, while disorder and impairment describe aspects of an organ's condition, the term disability describes aspects of a patient's condition.

Disorder refers to an abnormality in the anatomy or physiology of an organ and, in the case of a visual disorder, may occur anywhere in the visual system. Examples of visual disorders include corneal scar, cataract, macular degeneration, optic atrophy, or occipital stroke. It is important to recognize that knowing the specific visual disorder provides no information concerning the functional capacity of the eye.

Impairment refers to a functional abnormality in the organ. Thus, varying degrees of visual impairment can be measured in terms of specific visual functions, such as visual acuity, contrast sensitivity, visual field, or color vision. While such impairment measures demonstrate how well the eye functions, they do not reveal the impact of the visual disorder on the patient's ability to perform everyday activities. For example, a physician may state that the patient's visual acuity has dropped by four lines on the eye chart, while the patient reports an inability to see well enough to use a computer.

Disability refers to the ability of a patient (rather than an organ) to perform tasks, such as daily living skills, vocational skills, reading, writing, mobility skills, and so on. Since disability implies a broader perspective (the focus is on the person as a whole rather than on a specific organ), it is no longer entirely vision specific. For instance, while computer skills may be reduced due to vision loss, they may also suffer due to such conditions as arthritis. It is the combination of visual and nonvisual skills that determines the abilities or disabilities of an individual. Vision substitution techniques (such as the use of a white cane and increased reliance on memory and on hearing) may be helpful in improving the ability of an individual to perform specific tasks.

A disorder may cause impairments, and impairments may cause disability. However, these links are not rigid. An analysis of these various dimensions of vision loss permits identification of interventions at each link, which may improve the functional status and quality of life of an individual with visual loss (Figure 38.1; Fletcher 1999). For example, if one were interested in improving the ability of an individual with vision loss to use a computer effectively, possible interventions include medical and surgical intervention to impact the visual disorder and impairment, visual aids and adaptive devices to impact visual impairment, and social interventions, training, counseling, and education to impact visual disability. The design of a computer interface that enhances the ability of an individual with vision loss to perceive graphical and textual information would have a beneficial effect on the degree of impairment and resulting level of disability of that individual.

38.2.2 Epidemiology

Low vision has been defined as a permanent visual impairment that is not correctable with glasses, contact lenses, or surgical intervention, and which interferes with normal everyday functioning (Mehr and Freid 1975). Specifically, low vision is defined as a best-corrected visual acuity worse than 20/40 in the better-seeing eye (The Eye Diseases Prevalence Research Group 2004a). Blindness is defined as a best-corrected visual acuity of 20/200 or worse according to the U.S. definition and is defined as a best-corrected visual acuity of worse than 20/400 according to the World Health Organization definition (The Eye Diseases Prevalence Research Group 2004a). Based on demographics from the 2000 U.S. Census, an estimated 937,000 (0.78%) Americans older than 40 years were blind (U.S. definition); an additional 2.4 million Americans (1.98%) has low vision (The Eye Diseases Prevalence Research Group 2004a). Thus, blindness or low vision affects approximately 1 in 28 Americans older than 40 years (The Eye Diseases Prevalence Research Group 2004a). Largely due to the aging of the U.S. population, it is estimated that the number of blind persons older than 40 years in the United States will increase by approximately 70% to 1.6 million (prevalence of 1.1%) by the year 2020 (The Eye Diseases Prevalence Research Group 2004a). Vision loss has been ranked third, behind arthritis and heart disease, among conditions that cause persons older than 70 years to need assistance in activities of daily living (LaPlante 1988).

38.2.3 Selected Visual Disorders

In the United States, the most common causes of decreased vision are age-related macular degeneration (AMD), diabetic retinopathy, and glaucoma.

38.2.3.1 Age-Related Macular Degeneration

AMD is the leading cause of irreversible visual loss in the western world in individuals over 60 years of age. The macula is the part of the retina that is responsible for central vision. AMD affects more than 1.75 million individuals in the United States, and due to the rapid aging of the U.S. population, this number is estimated to increase to almost 3 million by the year 2020 (The Eye Diseases Prevalence Research Group 2004b). The prevalence of severe visual loss due to AMD increases with age. In the United States, at least 10% of persons between the ages of 65 and 75 years have lost some central vision due to AMD; among individuals over the age of 75 years, 30% have vision loss due to AMD.

Risk factors for this disease and its progression include age, sunlight exposure, smoking, ocular and skin pigmentation, elevated blood pressure, and elevated serum cholesterol levels. Recently, race has also been identified as a risk factor since AMD has been found to be more prevalent among whites than black persons (Bressler et al. 2008). Both Hispanics and Asians have a lower rate of AMD than whites, but still a higher rate of prevalence than blacks (Klein et al. 2006). The role of nutrition has not been fully identified as a risk factor, but a diet low in antioxidants and lutein may be a contributing factor.

AMD is a bilateral disease in which visual loss in the first eye usually occurs at about 65 years of age; the second eye becomes involved at the rate of approximately 10% per year. The two main types of AMD are atrophic and exudative. The atrophic (dry) form of the disease is generally a slowly progressive disease that accounts for approximately 90% of cases. It is characterized by the deposition of abnormal material beneath the retina (drusen) and by degeneration and atrophy of the central retina (also known as the macula); patients typically note slowly progressive central visual loss. Although much less common, the exudative (wet) form of the disease is responsible for about 88% of legal blindness attributed to AMD. This form of the disease, which often occurs in association with atrophic AMD, is characterized by the growth of abnormal blood vessels beneath the central retina (macula); these abnormal blood vessels elevate and distort the retina and may leak fluid and blood beneath or into the retina. Vision loss may be sudden onset and rapidly progressive (in contrast to the atrophic form of the disease, where vision loss generally occurs progressively over several months or years). AMD can cause profound loss of central vision, but the disease generally does not affect peripheral vision, and therefore, patients typically retain their abilities to ambulate independently.

Currently, no proven treatment reverses the retinal damage that has already occurred due to AMD. In order to try to prevent further vision loss, recommendations made to patients include eye protection against ultraviolet light exposure (sunglasses with ultraviolet light protection), no smoking, optimal control of blood pressure and serum cholesterol level, and a diet rich in dark green leafy and orange vegetables (antioxidants are believed to reduce the damaging effects of light on the retina through their reducing and free-radical scavenging actions; lutein is a macular pigment). The Age-Related Eye Disease Study (AREDS) was a randomized, controlled clinical trial that demonstrated a statistically significant reduction in rates of at least moderate visual acuity loss in persons with moderate AMD who received supplementation with vitamins C and E, beta carotene, and zinc compared to persons with moderate AMD assigned to placebo (Age-Related Eye Disease Study Research Group 2001). Laser photocoagulation and photodynamic treatment of the abnormal blood vessels found in patients with exudative macular degeneration may help to prevent severe vision loss in some cases. Intravitreal injections of pegaptanib, a pegylated modified oligonucleotide that binds to extracellular vascular endothelial growth factor (VEGF) isoform 165 (the isoform widely considered to be the primary pathologic form of VEGF), are associated with a higher likelihood of visual preservation and a slowing of visual loss among patients with exudative AMD (Gragoudas et al. 2004). Surgical rotation of the retina away from the area of abnormal blood vessels has also been effective in some cases. Other treatment modalities currently under investigation include such drugs as corticosteroids, other anti-VEGF agents, and combination treatments.

38.2.3.2 Diabetic Retinopathy

Approximately 16 million Americans suffer from diabetes mellitus, most of whom will develop diabetic retinopathy within 20 years of their diagnoses. In fact, after 20 years of diabetes, nearly 99% of those with insulin-dependent diabetes mellitus and 60% with non-insulin-dependent diabetes mellitus have some degree of diabetic retinopathy. Diabetic retinopathy is the leading cause of legal blindness in Americans aged 20–65 years, with 10,000 new cases of blindness annually. One million Americans have proliferative diabetic retinopathy, and 500,000 have macular edema. Among an estimated 10.2 million American adults aged 40 years and older known to have diabetes mellitus, the estimated crude prevalence rates for retinopathy and vision-threatening retinopathy are 40.3% and 8.2%, respectively (The Eye Diseases Prevalence Research Group 2004b). The estimated U.S. general population prevalence rates for retinopathy and vision-threatening retinopathy are 3.4% (4.1 million persons) and 0.75% (899,000 persons), respectively (The Eye Diseases Prevalence Research Group 2004c).

The major risk factor for diabetic retinopathy is duration of diabetes; it is estimated that at 15 years, 80% of diabetics will have background retinopathy and that, of these, 5%–10% will progress to proliferative changes. Other risk factors include long-term diabetic control (as reflected in serum levels of glycosylated hemoglobin), hypertension, smoking, and elevated serum cholesterol.

There are two main types of diabetic retinopathy: nonproliferative and proliferative. Nonproliferative diabetic retinopathy refers to retinal microvascular changes that are limited to the confines of the retina and include such findings as microaneurysms, dot and blot intraretinal hemorrhages, retinal edema, hard exudates, dilation and bleeding of retinal veins, intraretinal microvascular abnormalities, nerve fiber layer infarcts, arteriolar abnormalities, and focal areas of capillary nonperfusion. Nonproliferative diabetic retinopathy can affect visual function through two mechanisms: intraretinal capillary closure resulting in macular ischemia, and increased retinal vascular permeability resulting in macular edema. Clinically significant macular edema is defined as any one of the following: (1) retinal edema located at or within 500 μm of the center of the macula; (2) hard exudates at or within 500 μm of the center if associated with thickening of adjacent retina; and (3) a zone of retinal thickening larger than one optic disc area if located within one disc diameter of the center of the macula.

Proliferative diabetic retinopathy is characterized by extraretinal fibrovascular proliferation; that is, fibrovascular changes that extend beyond the confines of the retina and into the vitreous cavity. Fibrovascular proliferation in proliferative diabetic retinopathy may lead to tractional retinal detachment and vitreous hemorrhage. High-risk proliferative diabetic retinopathy is defined by any combination of three of the four following retinopathy risk factors: (1) presence of vitreous or preretinal hemorrhage; (2) presence of new vessels; (3) location of new vessels on or near the optic disc; and (4) moderate to severe extent of new vessels (Diabetic Retinopathy Study Research Group 1979).

Management of diabetic retinopathy includes referring the patient to an internist for optimal glucose and blood pressure control. In the Early Treatment Diabetic Retinopathy Study, focal or grid laser photocoagulation treatment for clinically significant macular edema reduced the risk of moderate visual loss, increased the chance of visual improvement, and was associated with only mild loss of visual field (Early Treatment Diabetic Retinopathy Study Research Group 1995). Intravitreal triamcinolone acetonide and other intravitreal anti-VEGF agents are currently under investigation for the treatment of diabetic macular edema. Panretinal laser photocoagulation treatment of eyes with high-risk proliferative diabetic retinopathy reduced the risk of severe visual loss by 50% compared to untreated control eyes (Diabetic Retinopathy Study Research Group 1981). Surgery is often indicated for nonclearing vitreous hemorrhage and for tractional retinal detachment involving or threatening the macula.

38.2.3.3 Glaucoma

Primary open angle glaucoma (POAG) is the most prevalent type of glaucoma, affecting 1.3–2.1% of the general population over the age of 40 years in the United States. In the United States, the disease is the leading cause of irreversible blindness among blacks and the third leading cause among whites (following AMD and diabetic retinopathy), and is responsible for 12% of legal blindness. Risk factors for the disease include increasing age (especially greater than 40 years), African ethnicity, positive family history of glaucoma, diabetes mellitus, and myopia (nearsightedness).

POAG is a chronic, slowly progressive optic neuropathy characterized by atrophy of the optic nerve and loss of peripheral vision. Central vision is typically not affected until late in the disease. Because central vision is relatively unaffected until late in the disease, visual loss generally progresses without symptoms and may remain undiagnosed for quite some time. While usually bilateral, the disease may be quite asymmetrical. POAG is associated with increased intraocular pressure, but normal-tension glaucoma may cause glaucomatous vision loss in patients with normal intraocular pressure. Thus, normal eye pressure does not rule out the presence of glaucoma.

Treatment of POAG includes topical or systemic medications, laser, or surgery to lower the intraocular pressure to a level at which optic nerve damage no longer occurs. Visual field testing is performed regularly in order to evaluate for progressive loss of peripheral vision, and the optic nerve is examined regularly to evaluate for evidence of progressive optic atrophy (clinical signs of glaucoma in the optic disc include asymmetry of the neuroretinal rim, focal thinning of the neuroretinal rim, optic disc hemorrhage, and any acquired change in the disc rim appearance or the surrounding retinal nerve fiber layer).

38.2.4 Specific Visual Functions

38.2.4.1 Visual Acuity

Visual acuity is the most common measure of central visual function and refers to the smallest object resolvable by the eye at a given distance. It is defined as the reciprocal of the smallest object size that can be recognized. Visual acuity is expressed as a fraction in which the numerator is the distance at which the patient recognizes the object and the denominator is the distance at which a standard eye recognizes the object. For instance, a visual acuity of 20/60 means that the patient needs an object three times larger or three times closer than a standard eye requires. The traditional visual acuity chart presents symbols of decreasing size with fixed high contrast. The visual acuity chart used most often in the clinical setting is the Snellen acuity chart, which is comprised of certain letters of the alphabet; the size of the letters are constant on a given line of the eye chart, and decrease in size the lower the line on the chart. In accurate Snellen notation, the numerator indicates the test distance and the denominator indicates the letter size seen by the patient.

38.2.4.2 Contrast Sensitivity

Contrast sensitivity refers to the ability of the patient to detect differences in contrast and is defined as the reciprocal of the lowest contrast that can be detected. This may be measured with the Pelli-Robson chart, in which letters decrease in contrast rather than size, or the Bailey-Lovie chart, in which letters of a fixed low contrast are varied in size. Contrast sensitivity is considered a more sensitive indicator of visual function than Snellen acuity and may provide earlier detection of such pathology as retinal and optic nerve disease.

38.2.4.3 Visual Field

Visual field is classically defined as a three-dimensional graphic representation of differential light sensitivity at different positions in space. Perimetry refers to the clinical assessment of the visual field. Typically, visual field is assessed with kinetic or static perimetry. During kinetic perimetry, a test object of fixed intensity is moved along several meridians toward fixation and points where the object is first perceived are plotted in a circle. During static perimetry, a stationary test object is increased in intensity from below threshold until perceived by the patient, and threshold values yield a graphic profile section. While peripheral visual field loss often produces difficulty for patients in orientation and mobility functions, macular field loss (either centrally or paracentrally) often causes difficulties with reading. For instance, the presence of central or paracentral visual field loss is a more powerful predictor of reading speed than is visual acuity (Fletcher et al. 1994).

38.2.4.4 Color Vision

Evaluation of color vision may be performed using pseudoisochromatic color plates, which are quick and commonly available; they consist of circles in various colors such that a person with normal color vision function can distinguish a number from the background pattern of circles. Ishihara or Hardy-Rand-Rittler pseudoisochromatic color plates are designed to screen for congenital red/green color deficiencies, while Lanthony tritan plates may be used to detect blue/yellow defects, which are frequently present in acquired disease. With the Farnsworth-Munsell 100-hue test, the patient must order 84 colored disks; the time-consuming nature of this test limits its clinical use. The Farnsworth Panel D-15 is a shorter and more practical version (using 15 disks), but is less sensitive. Most color-vision defects are nonspecific.

38.2.5 Visual Function and Age

This section provides an overview of how levels of visual function vary with age, and to what degree. Aging is synonymous with natural declines in a person's sensory abilities. As such, the process of aging is accompanied by changes to the eye, including the retina and visual nervous system that can impact functional vision (Schieber 1994). Additionally, older adults are more likely to acquire ocular conditions that can compromise visual functioning beyond normally anticipated changes, such as macular degeneration, diabetic retinopathy, and cataracts. Age-related vision loss commonly impinges on the ability to complete near vision tasks such as reading and using the computer (Arditi 2004). An understanding of these functional declines provides direction for strategies aimed to mitigate the negative impact of these changes. HCI designers, developers, and usability specialists should be fully aware of these needs, as the needs of this growing user population will become an increased priority with the shift in demographics of population segment.

Aspects of visual function that are known to normally decline as part of the aging process include the following:

- Visual acuity
- Visual field
- Contrast sensitivity
- Color perception
- Floaters
- Dry eyes
- Increased need for light
- Difficulty with glare
- Dark/light adaptation
- Reduced depth perception (Orr 1998)

Beyond these factors, eye movement efficiency and accuracy are observed to decline with old age. Older adults are known to be less accurate and/or slower in locating a target in the peripheral vision (see also Kline and Scialfa [1997]; Lee, Legge, and Oritz [2003]). Age-related differences have also been observed with the effectiveness with which older adults visually track targets with higher velocities. In both cases, these trends are typically aggravated by the presence of distracting stimuli (visual, auditory, tactile, etc.) in the background or foreground that contribute to the complexity. The perception of moving stimuli, for older adults, is both less effective and less efficient in tasks aimed at the detection of small target movement/change such as those found on dials and controls (Kline and Scialfa 1997). Furthermore, deficits in central, paracentral, and peripheral visual field can pose different demands on vision, resulting in different search strategies related to eye movements (Coeckelbergh et al. 2002). Schumacher and colleagues (2008) examined the reorganization of visual processing and its relation to eccentric viewing in patients with macular degeneration, revealing that visual stimulation of the preferred retinal location (PRL) in patients with AMD increased brain activity in the cortex normally representing central vision relative to visual stimulation of a peripheral region outside the patients' PRL and relative to stimulation in the periphery of age-matched control participants.

Older adults tend to exhibit a greater degree of difficulty with visual search tasks, especially when the number of items to be searched increases (Kline and Scialfa 1997). A recent study explored the hypothesis that older adults are slower due to a greater degree of double checking during visual search than younger adults. The research showed that older adults did double check more often, however speed stress instructions reduced age-related differences in double checking (Mitzner et al. 2010). Older adults have a propensity for longer visual reaction times, especially in cases where attention is divided. Furthermore, this population segment experiences difficulties ignoring extraneous information, or background noise (Schieber 1994). Research also suggests that visual search is slower and less effective for older adults due to a shrinking of the useful field of view to which attention can be simultaneously allocated. The size of the useful field of view, for older adults, is especially susceptible to context-related factors, such as complexity and cognitive task load (Schieber 1994).

38.2.6 Summary

Studies have demonstrated that ophthalmic patients are at high risk for decreased functional status and quality of life (Parrish et al. 1997; Rovner et al. 2011; Scott et al. 2001; Scott et al. 1994). Patients' functional statuses and qualities of life may be improved by interventions that increase visual function, such as surgery to repair retinal detachment or remove epiretinal membrane (Scott et al. 1998) and surgery to remove cataract (Applegate et al. 1987; Brenner et al. 1993; Donderi and Murphy 1983; Javitt et al. 1993; Steinberg et al. 1994). In addition, functional status and quality of life may be improved by interventions, such as low vision devices and services, which permit patients to use their remaining vision more effectively (Scott et al. 1999; Stelmack et al. 2008). In addition, as the proportion of older adults multiplies, the number of individuals experiencing some degree of vision dysfunction that normally occurs with age has created an increased demand for information technology that affords use despite the dysfunction. Prior studies have demonstrated the effect of low vision interventions on objective task-specific measures of functional abilities such as reading speed, reading duration, and ability to read a certain print size (Nilsson 1990; Nilsson and Nilsson 1986; Nguyen, Weismann, and Trauzettel-Klosinski 2009; Rosenberg et al. 1989; Sloan 1968).

However, historically there has been little data available concerning the abilities of people with visual impairments to use computers. It has been even more challenging to find data concerning how modifications of GUI features may increase accessibility of computers to patients with visual impairments. An exception to this is the research agenda established by Jacko and colleagues (see Barreto, Jacko, and Hugh [2007]; Jacko et al. [2005]; Jacko et al. [2003]; Scott, Feuer, and Jacko [2002a,b]; Scott et al. [2006]; and Table 38.1, e.g., has yielded systematically derived HCI performance thresholds for users according to their ocular profile [diagnoses and functional ability]). Table 38.1 summarizes the research products resulting from this effort, chronologically organized by the ocular pathology investigated, the age of the users involved, the GUI interaction investigated and the corresponding specific interface feature in question. It goes without saying that the rapid proliferation of visual displays beyond the desktop to handheld and wearable computers and other mobile devices, such as phones and tablets, increase the importance of GUI innovations that facilitate efficient and rewarding usage by people with visual impairments.

38.3 HIGHLIGHTING TECHNOLOGICAL ADVANCEMENTS IN HUMAN–COMPUTER INTERACTION RESEARCH

With nearly every aspect of today's society involving some type of computer technology, there is an ever-growing need to understand how individuals with visual impairments can access technology. Currently, without special modifications, the typical PC poses several challenges to users who experience limited bandwidth to their visual sensory channels, as well as other perceptual impairments. Furthermore, the widespread mainstream adoption of technologies that leverage touch screen displays, such as MP3 players and smart phones, has introduced novel interaction methods that pose new constraints to use for those with visual impairments. As a result, the HCI research community is placing great emphasis on the design of universally acceptable technologies. According to Stephanidis et al. (1998, p. 6), "Universal access in the Information Society signifies the right of all citizens to obtain equitable access to, and maintain effective interaction with, a community-wide pool of information resources and artifacts." Accessibility has been a term traditionally associated with elderly individuals, individuals with disabilities and others who possess special needs (Stephanidis et al. 1999). However, because of the current influx of new technologies into the market, the population of users who may possess special needs is growing. As a result, accessibility has taken on a more comprehensive connotation. This connotation implies that all individuals with varying levels of abilities, skills, requirements, and preferences be able to access information technologies (Stephanidis et al. 1999). Universal access also implies more than just adding features to existing technologies. Rather, the concept of universal access emphasizes that accessibility be incorporated directly into the design (Stephanidis et al. 1998). Perceptual and adaptive interfaces are two ideal examples of how universal accessibility can be achieved.

38.3.1 Perceptual Interfaces

The concept of perceptual design describes a perspective of design that defines interactions in terms of human perceptual

TABLE 38.1

Visualization of the Breadth and Depth of the Investigations, To-Date, for the Derivation of HCL Performance Thresholds for Users According to Their Ocular Profile (Diagnoses and Functional Ability)

Investigation			Ocular Pathology			Age			GUI Interaction						GUI Interface Feature							
Year	Authors	None	AMD	Diabetic Retinopathy	Other	18–55 years	55 + years	Visual Search	Icon Selection	Cursor Movement	Drag & Drop	Menu Selection	Distr-acters	Icon Size	Set Size	Back-ground	Visual Profiles	Tactile Cues	Auditory Cues	Text Size	Desk-top PC	Hand-help PC
1998	Jacko & Sears		•	•	•	•	•	•	•				•	•	•	•	•				•	
1999	Jacko		•	•	•	•	•	•	•				•	•	•	•	•				•	
1999	Jacko, Dixon, et al.		•	•	•	•	•	•	•				•	•	•	•	•				•	
2000	Jacko, Barreto, et al.		•	•	•	•	•	•	•				•	•	•	•	•				•	
2000	Jacko, Rosa, et al.		•				•	•		•			•				•				•	
2001	Jacko et al.		•				•	•	•				•	•	•	•	•				•	
2002[a]	Scott, Feuer, & Jacko		•				•	•	•				•	•	•	•	•				•	
2002[b]	Scott, Feuer, & Jacko		•				•	•	•				•	•	•	•	•				•	
2002	Jacko, Barreto, et al.		•				•	•	•				•	•	•	•	•				•	
2002	Vitense, Jacko, et al.		•				•				•						•	•	•		•	
2003	Jacko, Vitense, & Scott	•				•		•			•		•					•	•		•	
2003	Vitense, Jacko, & Emery	•				•		•			•		•					•	•		•	
2003	Emery, Edwards, et al.	•					•				•							•	•		•	
2003	Jacko, Scott, et al		•				•				•						•	•	•		•	
2004	Jacko, Barnard, et al.		•				•				•						•	•	•		•	
2004	Jacko, Emery, et al.	•					•				•							•	•		•	
2004	Edwards, Barnard, et al.			•		•	•	•				•	•	•		•	•	•	•	•	•	
2005	Edwards, Barnard, et al.			•		•	•	•				•	•	•		•	•	•	•	•	•	
2005	Jacko, Moloney, et al.		•				•				•						•	•	•		•	
2005	Moloney, Leonard, et al.		•				•				•						•	•	•		•	
2005	Leonard, Edwards, & Jacko			•		•	•	•				•	•	•		•	•	•	•	•	•	
2005	Leonard, Jacko, & Pizzimenti		•					•	•	•	•		•	•	•	•	•	•	•	•		•
2006	Scott et al.		•				•				•						•		•		•	
2006	Moloney, Shi, et al.		•				•				•						•	•	•		•	

Note: • indicates that a given topic is addressed by the corresponding investigation, listed in the leftmost column.

capabilities. In a sense, it strives to humanize interaction. The design of perceptual interfaces adheres to the idea that lessons learned from psychological research about perception can be applied to interface design (Reeves and Nass 2000). In adopting the concept of perceptual design, several opportunities surface for the creation of innovative perceptual user interfaces. Interactions with these interfaces can be described in terms of three particular human perceptual capabilities; chemical senses (e.g., taste and olfaction), cutaneous senses (e.g., skin and receptors), and vision and hearing (Reeves and Nass 2000). Although commonly used computer technology limits the effectiveness of chemical senses, the technology can be extended to incorporate the cutaneous, visual, and hearing senses.

In terms of vision, research has focused on topics, such as visual mechanics, color, brightness and contrast, objects and forms, depth, size, and movement. Hearing research includes psychophysical factors such as loudness, pitch, timbre, and sound localization; physiological mechanisms such as the auditory components of the ear, and the neural activity associated with hearing; and the perception of speech such as units of speech and the mechanics of word recognition (Reeves and Nass 2000). With respect to the cutaneous senses, augmented GUIs with haptic feedback have been around since the early 1990s. Akamatsu and Sate conducted the first research with a haptic mouse that produced haptic feedback via fingertips and force feedback via controlled friction (as cited in Oakley et al. 2000). Engle, Goossens, and Haakma found that directional two degrees of freedom force feedback improved speed and error rates in a targeting task (as cited in Oakley et al. 2000).

The strength of perceptual user interfaces comes from the ability of designers to combine an understanding of natural human capabilities with computer input/output devices, and machine perception and reasoning (Turk and Robertson 2000). General examples of how capabilities can be combined with technology include speech and sound recognition and generation, computer vision, graphical animation, touch-based sensing and feedback, and user modeling (Turk and Robertson 2000).

From an applied research standpoint, the concepts of perceptual interfaces are housed within multimedia and multimodal interfaces. Both multimedia and multimodal interfaces offer increased accessibility to technologies for individuals with perceptual impairments. Distinctions can be drawn between perceptual, multimedia, and multimodal interfaces, shown in Figure 38.2. Perceptual interfaces prescribe human-like perceptual capabilities to the computer. Multimedia and multimodal interfaces can be considered applied extensions of this concept. Multimedia interfaces elicit perceptual and cognitive skills to interpret information presented to the user. Whereas multimodal interfaces use multiple modalities for the HCI. Multimedia interfaces focus on the media while multimodal interfaces focus on the human perceptual channels (Turk and Robertson 2000). The strength and capabilities of multimedia and multimodal interfaces with respect to individuals with perceptual impairments are described in more depth in Sections 38.3.1.1 and 38.3.1.2.

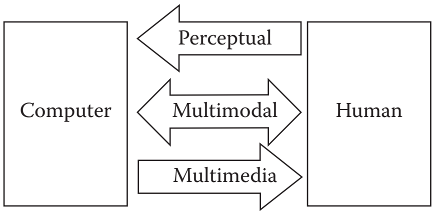


FIGURE 38.2 Perceptual, multimodal, and multimedia interfaces (flow of information).

38.3.1.1 Multimedia Interfaces

Multimedia interfaces have grown from the need to display diverse forms of information in a flexible and interactive way. Multimedia can be simply defined as computer-controlled interactive presentations (Chignell and Waterworth 1997). The broadness of this definition directly corresponds to the broadness of the field of multimedia research. Chapter 6 in this book contains a comprehensive discussion of multimedia including topics dealing with the cognitive implications related to multimedia, selecting media for the message, navigation and interaction and evaluation of multimedia. However, for the purpose of this chapter, the discussion of multimedia is limited to a brief overview of types and potential strengths of multimedia as they relate to enhancing accessibility to information technologies for individuals with perceptual impairments.

There are three approaches to multimedia: performance, presentation, and document (Chignell and Waterworth 1997). In the performance approach, multimedia is a kind of theatrical play that is conveyed through “actors.” The timing of the actors’ performances is orchestrated in an effort to entertain and educate (Waterworth and Chignell 1997). Presentation multimedia is a modern version of slide shows, where video clips and animation enhance a sequence of slides. The goal of the presentation approach is to convey ideas to the user (Chignell and Waterworth 1997). Lastly, the document approach focuses on text and ideas. It can be thought of as an enhanced document that elaborates ideas in the text. All of these approaches provide additional opportunities to convey perceptual information to the user.

Multimedia allows communication between users and computers in a sensory manner. As such, the essential aspect of designing multimedia interfaces is selecting the media and modalities. Information can be taken from one modality and presented in another modality (Chignell and Waterworth 1997). For instance, data presented visually could be converted and displayed audibly. For instance, multimedia could be used to enhance a GUI for a user with a hearing impairment. Information that would be commonly conveyed via auditory feedback could be provided visually. An example of this would be to have an icon display when an error has been committed, rather than the traditional beep sounding. Other examples of how multiple modalities can enhance accessibility to information technologies are discussed in the Section 38.3.1.2.

The potential strength of multimedia interfaces comes from its ability to use images, text, and animation to connect with users. However, in order to facilitate that type of exchange, much consideration must be paid to users' sensations and perceptions via the auditory, speech, and visual channels. For instance when designing graphical images, basic knowledge of color vision is needed to ensure that colors can be discriminated, foreground can be separated from background, highlights will attract attention, and grouping of objects is apparent (Gillan 1998). An understanding of human sensation and perception is necessary if perpetual interfaces are to reach their full potentials. When perceptual interfaces are used, however, they provide individuals with perpetual impairments, such as those related to the visual channel, alternative modalities to access technology. Multimodal interfaces are a second type of perceptual interface that provide this same benefit.

38.3.1.2 Multimodal Interfaces

Multimodal interfaces, as they are discussed in this chapter, are interfaces that support a wide range of perceptual capabilities (e.g., auditory, speech, and visual) as a means to facilitate human interaction with computers. Multimodal interfaces are discussed in a much broader sense (e.g., development of multimodal processing) in Chapter 18.

With the growing complexity of technology and applications, a single modality no longer permits users to interact effectively across all tasks and environments (Oviatt, Cohen, Suhm, et al. 2002). The strength of a multimodal design is its ability to allow users the freedom to use a combination of modalities or the best modality for their needs. These interfaces make the most effective use of the variety of human sensory channels, alone and in combination. Ultimately, multimodal interfaces offer expanded accessibility of computing and promote new forms of computing not previously available to individuals with perceptual impairments.

The development and application of multimodal interfaces for making technologies accessible to users with visual impairments is growing. Some forms of unimodal, bimodal, and trimodal feedback within multimodal interfaces have been investigated and their advantages have been documented for users with visual impairments (Jacko, Emery, et al. 2004; Jacko, Barnard, et al. 2004; Jacko et al. 2005; Jacko et al. 2003). Since most information presented on a GUI is visual, there is great interest in the research community to find alternative ways of displaying this information. One of the common approaches to conveying visual information in a nonvisual way is through the use of the tactile modality. Specific research has been conducted in the realm of tactile displays with respect to visual impairments. Not only can tactile displays provide information regarding a graphic's identity but also the depth, location, and perception of its purpose. The use of tactile systems can also provide navigational information. Research has shown that tactile output of directional information offers support to the blind as they explore images (Kurze 1998). Research has also been conducted in the area of movable dynamic tactile displays that present information to one or several fingertips in a Braille type manner. The Braille dots move in a wave of lifted and lowered series of pins (Fricke and Baehring 1994). Along the same line of research, the use of a bidimensional single cell Braille display combined with a standard Braille cell has been evaluated. Although initial research found this new combined device is not an improvement over a standard stand-alone Braille display, continued research is yielding improvement (Ramstein 1996). Tactile output via force feedback has been looked to as a means of conveying numerical information. Yu, Ramloll, and Brewster (2000) worked on a system that converts data typically displayed visually into haptic and auditory output. Since data visualization techniques are not appropriate for blind people or people with visual impairments, this system translated graphs into friction and textured surfaces along with auditory feedback. Pins presented via a tactile mouse to communicate graphical interface objects have been shown to support users in performing web-based tasks with fewer limitations than using a screen-reader alone (Kuber, Yu, and O'Modhrain 2010). Another common approach to converting visual information in a nonvisual way is through the use of the speech modality. Speech recognition systems serve as an alternative modality for users and computers to interact. These systems recognize human speech and translate it into commands or words understood by the computer. Chapter 40 in this handbook offers an introduction to the technology of speech recognition systems and design issues associated with incorporating speech recognition into applications. However, this chapter's discussion of speech recognition systems concentrates on its use related to Individuals with visual impairments. The integration of speech input and output into applications offers an alternative to purely graphical environments (Yankelovich, Levow, and Marx 1995). This type of technology-driven design allows applications to be suited to a wide variety of individuals with disabilities (Danis and Karat 1995). For instance, individuals with visual impairments can use a computer solely by voice activation. A multimodal web browser is another approach for individuals with visual impairments to access the web. MGSYS VISI-VL is designed to couple tactile keyboard access with the speech recognition process (Ismail and Zaman 2010).

Speech recognition systems are traditionally associated with the concept of dictation. Products such as Dragon Systems, Inc.'s Dragon Naturally Speaking offer a line of speech recognition products for dictation. Some specific packages are geared toward particular professions, such as medical and legal (Cunningham and Coombs 1997). Other common dictation systems are IBM ViaVoice, Lernout & Hauspie Voice Xpress, and Philips Speech Processing FreeSpeech98. Along the lines of recognition engines, Verbex Voice Systems and SRI Corp.'s DECIPHER offer continuous speech recognition technology. Microsoft's Whisper provides speaker-independent speech recognition with online adaptation, noise robustness, and dynamic vocabularies and grammars (Huang, Acero, Alleva, et al. 1995). Speech-driven menu navigation systems, for instance Command Corp. Inc.'s IN CUBE Voice Command for window navigation, have also been developed

(Karshmer, Brawner, and Reiswig 1994; Karshmer et al. 1994).

Often, circumstances may affect more than one sensory channel, and individuals may experience multiple impairments. In addition to providing heightened challenges to interaction, this impacts the efficacy of perceptual interface in meeting users' needs. For example, many disabilities are associated to some degree with speech degeneration, and therefore speech recognition systems may not fully accommodate a user's needs (Rampp 1979). Older adults commonly experience multiple impairments, as all the sensory channels normally decline with age, and are more susceptible to conditions, such as stroke, which can cause reduced speech, vision, and motor skills. For cases in which speech is degraded, speech recognition systems must be sensitive enough to adapt to impaired speech. Regardless of what caused the speech impairment, devices and techniques can be applied to augment the communicative abilities of individuals who experience difficulty speaking in an understandable manner.

Augmentative and alternative communication (AAC) is a field of study concerned with providing such devices and techniques. McCoy et al. (1997) described a prototype system aimed at users with cognitive impairments. This prototype is designed to aid communication and provide language intervention benefits across several user populations. An iconic language approach has been applied to aid individuals with significant speech and multiple impairments (SSMI). Research has been conducted on the use of icon language design based on the theory of icon algebra (Chang 1990) and the theory of conceptual dependency (Schank 1972). These methodologies are then used in interactive interface design (Albacete, Chang, and Polese 1994). Individuals with SSMI also commonly have difficulty with word processing. Using an animated graphical display, phoneme probabilities of speech can be more easily isolated and recognized. This allows users the opportunity to interpret their speech rather than forcing the computer to do it automatically (Roy and Pentland 1998). The more feedback and modalities users are provided, the more efficiently they can interact with speech recognition by assisting the computer to interpret their speech correctly. Speech recognition systems must also take into account the possible extent of cognitive burden placed on the user. If voice is used to navigate through a GUI, certain prosodic features (e.g., pauses) of the user's speech, resulting from a high cognitive load, may affect performance (Baca 1998). Additional challenges associated with the implementation of speech recognition systems are discussed in this handbook as well. Overall, these systems need to account for situations when users with visual impairment experience varying levels of hearing, speech, and cognitive abilities.

The auditory channel serves not only to convey information, but also to receive information. The use of auditory feedback is extremely useful to many computer users. Ongoing research has looked at the use of bidirectional sound as a standard element of an interface environment. A prototype named the Voice Enabled Reading Assistant (VERA), written using an aural-oral user interface (A-OUI) model, was developed to provide bidirectional sound. The A-OUI model captures qualities and functions of plain text files needed for user interfaces to present information to the auditory, visual, and tactile senses. The use of the VERA prototype can be applied to many types of office and Internet text and data. Similar speech-enabled products are Emacspeak and IBM VoiceType Simply Speaking Gold (Ryder and Ghose 1999).

Much work with respect to audio feedback has focused on conveying information from a GUI. For instance, research conducted by Darvishi and colleagues (Darvishi, Guggiana, et al. 1994; Darvishi, Munteanu, et al. 1994) looked at mapping GUIs into auditory domains through impact sounds based on physical modeling. Ultimately, these sounds are used to convey information to the user regarding the objects in a GUI environment. Audio feedback has also been used to provide visually impaired users with a sense of depth perception, by varying the location of the sound sources in a three-dimensional environment. Because depth perception is a function of vision, any cues that can be conveyed though other modalities are vital. This is particularly crucial when working with three-dimensional computer applications, such as a CAD package (Mereu and Kazman 1996). The use of auditory feedback has also been studied with regard to enhancing synthesis speech output. Through the development of spatial audio processing systems, a greater benefit of synthesis speech was achieved (Crispien, Würz, and Weber 1994). IBM's Screen Reader/2 also uses synthesis speech to make GUIs accessible to those who are visually impaired or blind, by converting screen information to speech or Braille. The users are continuously kept informed of screen activity and cursor movement. Reading typed characters, words, and sentences are features that can be made automatic. This software aids in the use of windows, menus, dialog boxes, and other controls (Thatcher 1994). Microsoft Corporation's Whistler, a trainable, text-to-speech system that produces synthetic speech, sounds very natural by reproducing the characteristics of original speech (Huang, Acero, Adcock, et al. 1996). WRITE:OUTLOUD and OutSPOKEN are two additional text-to-speech products commonly used (Friedlander 1997). Regardless of what product is used, text-to-speech software enables information to be collected and utilized in a more rapid manner.

In addition to the benefits already discussed that enable users with visual impairments to access technology, multimodal interfaces also provide superior support of error handling. More specifically multimodal interfaces have been shown to have the ability to avoid errors and recover more efficiently from errors when they do occur. This is a very important element of universal accessibility. Enabling users to use a technology includes not only being able to access the technology but also the ability to use it in an efficient manner. Reducing or avoiding errors is a major key to improving efficiency. A multimodal interface provides better error handling than a unimodal interface for several reasons. The following are factors that represent reasons why a multimodal interface may be better at error avoidance:

- Users will select the input mode considered less error prone for a particular context, which is assumed to lead to error avoidance.
- The ability to use several modalities permits users the flexibility to leverage their strengths by using the most appropriate modality. It is possible that the more comfortable the users, the less likely they are to make errors.
- Users tend to switch modes after systems errors. For instance, users are likely to switch input modes when they encounter a system recognition error.
- Users are less frustrated with error when interacting within a multimodal interface, even when errors are as frequent as in a unimodal interface. The reduction in frustration may be the result of the user feeling as though they have more control over the system because they can switch modes (Oviatt 1999).

Ultimately, multimodal systems aid in reducing errors (error handling) because they offer parallel or duplicated functionality that allow users to accomplish the task using one of several modalities (Oviatt and Cohen 2000). These benefits stem from the user-centered and system-centered design perspectives from which multimodal interfaces are built (Oviatt and Cohen 2000; Oviatt, Cohen, Suhm, et al. 2002). Like multimodal interfaces, adaptive interfaces, also benefit from a user-centered perspective.

38.3.2 Adaptive Interfaces

Adaptive interfaces have great potential for accommodating a wide range of users in a variety of work contexts. As a result, much research has been conducted on the design and implementation of adaptive interfaces. An example of early research in the field of adaptive GUI is illustrated by the work of Mynatt and Weber (1994) with the Mercator project, and the GUIB (Textual and Graphical User Interfaces for Blind People) project (Petrie, Morley, and Weber 1995). Both of these projects focused on making environment-level adaptations to GUIs in order to make them more accessible (Stephanidis 2001a). Mercator interfaces model the graphical objects and their hierarchical relationships. The model serves to predict a user's interaction (Edwards and Mynatt 1994; Edwards, Mynatt, and Stockton 1994). The goal is to provide visually impaired users with an interface that is more accessible. By better understanding how low vision users interact with a computer, interfaces can be designed more effectively. Outputs such as synthetic and digitized speech, refreshable Braille, and nonspeech sounds also make an interface easier to use. Auditory icons and earcons are two predominate areas of nonspeech sound research. Auditory icons, developed by Gaver (1989), are everyday sounds that occur in the world mapped to the computer world. Gaver first applied the concept of auditory icons to SonicFinder. Earcons are a second method for employing nonspeech audio in GUIs. Earcons, developed by Blattner, Sumikawa, and Greenberg (1989), are audio messages that provide the user with information about objects or operations of the computer. The first version of GUIB adapted the GUI by combining Braille speech and nonspeech audio together to construct a nonvisual interface so that users who are blind can access the GUI (Emiliani 2001).

Configurable interface designs based on user models have been heavily researched. The user models have been defined with respect to visual, cognitive, motor, and other abilities. With these models, custom computer systems, including both hardware and software, can be created (McMillan and Wisniewski 1994). Semantic abstraction of user interaction, named abstract widgets, is another modeling approach that provides great flexibility. Abstract widgets separate the user interface from the application functionality. This allows users to interact with interfaces as they choose, independently from their environment (Kawai, Aida, and Saito 1996). The use of adaptation determinates, constituents, goals, and rules are yet another approach of an adaptation strategy. This strategy is based on the fact that these important attributes, which categorize adaptation, can be used to formulate adaptation rules. These adaptation rules, in turn, assist the development of intelligent user interfaces. This approach can be customized to the requirements of different application domains and user groups (Stephanidis, Karagiannidis, and Koumpis 1997). Approaches such as described by the pervasive accessible technology (PAT) allow individuals with disabilities to use standard interface devices that adapt to the user in order to communicate with information technology infrastructures (Paciello 1996). Based on an individual's disability, the implementation of a user interface management system (UIMS) model provides the versatility needed to adapt interfaces to individuals. The selection of input devices, presentation of information on the screen, and choice in selection/activation method, can all be adapted to fit specific user needs (Bühler, Heck, and Wallbruch 1994). Current research focuses on operationally reliable software infrastructures that support alternative physical realizations through the abstractions of objects. More specifically, systems such as Active X (by Microsoft) and JavaBeans (by SunSoft) represent component-ware technology (Stephanidis 2001b). These systems represent a mainstream effort to provide technological structures that provide more adequate support for accessibility. They are also two examples of currently available tools that can be used to generate code for the adaptation of various interface components.

Some hypermedia systems research has focused on adaptive hypermedia applications. These adaptive hypermedia systems keep track of evolving aspects of the user, such as preferences and domain knowledge. This information is stored to create a user model, which in turn is the basis for user interface adaptation (De Bra, Houben, and Wu 1999). Adaptive hypermedia can also be designed based on task models. In the latter case, task models are the basis from which hypermedia systems are developed. These task models support the design and development of hypermedia. Different task models are associated with different types of users (Paternó and Mancini 1998). Task models reflect the user's view of the activities to be performed. In addition to personalizing the content of hypermedia systems, adaptive navigation support has also been researched. Prototype systems have been developed to demonstrate how different navigational possibilities can be made available based on a user model (Pilar da Silva et al. 1998).

Another example of adapting interfaces to users is through the use of the EZ Access protocols developed by the Trace Center at the University of Wisconsin-Madison. EZ Access protocols are a set of techniques that modify an interface to fit a specific user's need. These protocols work across disabilities and with a range of products. The most common use of EZ Access protocols is in touchscreen kiosks (Vanderheiden 1998).

While research and development in the areas of perceptual and adaptive interfaces, like those examples discussed, have gained substantial attention in generating potential solutions for individuals with visual sensory impairments, the majority of work and realized products are positioned in two areas. The most readily available products and the majority of the research have been somewhat narrowly focused on (1) the magnification of screen elements (Fraser and Gutwin 2000) and (2) the accessibility of text (Craven 2003). Comparatively, less has been accomplished in terms of critical aspects of the GUI as has been done with magnification and vocalization of text. While the solutions generated have afforded access to both those individuals who are blind and those who retain a range of visual impairment, they tend to be one size fits all solutions, which entirely abandon the visual sensory channel (as in the case of Braille interfaces or Screen Readers), or focus only on the augmentation of the visual channel (e.g., magnification, increased contrast, increased text size, etc.), and very few are multimodal, or perceptual. Table 38.2 summarizes a selection of popular assistive devices. Following this table, a discussion of research that has evaluated the efficacy of such approaches to accessibility is presented.

Kline and Glinert (1995) presented UnWindows V1, a set of interface tools to support selective magnification of the window area, and tracking the location of the mouse pointer on the display screen. The authors noted, "Magnification is one method commonly employed to help low vision users deal with the small type fonts, illustrations, and icons present in much of today's printed media and computer displays" (Kline and Glinert 1995, p. 2). Key components of the UnWindows system included (1) a dynamic magnifier to compensate for the loss of global context imposed by static magnification and changing display content and (2) visual and aural feedback to aid the users in locating the mouse pointer. Kline and Glinert placed emphasis on the problematic nature of visual tracking in the presence of a screen densely populated with icons and windows. Interestingly, they received mixed reaction to their interface by users with and without visual impairment, especially in terms of the auditory feedback provided whenever the mouse pointer entered a new window (users found this annoying). While no formal empirical testing was performed in relation to UnWindows, questions surface as to the effectiveness of nonvisual, multimodal feedback in a complex display.

A usability review of currently available technologies for the conversion of GUI technology for use by individuals who were blind or possessed low vision, the ability of magnification, synthetic speech, and Braille were reviewed for their ability to provide the respective users 100% access to GUIs on nine test areas (Becker and Lundman 1998). These tests included

- Installing and configuring the device/software
- Uninstalling the device/software
- Performance reliability/stability

TABLE 38.2
Popular Assistive Devices

Competitor	Product Overview	Key Limitation(s)
Screen reader 	Software program that reads to the user elements that appear on an interface via synthesized voice. The program reads left to right, starting at the very top of the screen. When an image is encountered, the program reads the associated ALT text.	Solution abandons any remaining vision the user has, using only their auditory ability. Efficacy depends on the organization of the interface (e.g., anything not modeled in a left to right organization is not compatible).
Braille display 	Similar to the screen readers, but gives the reader the information via tactile cues (Braille characters).	Same limitations as a screen reader, plus the user has to learn Braille, which is not likely if they have residual vision, and are losing vision later in life.
Screen magnifier 	Physical device or software program that enlarges the entire screen image. Software programs, such as Zoomtext, developed by Ai squared, allow for "adaptable" magnification of the interface, and some versions incorporate a screen reader.	A "one size-fits all" solution, that is not adaptable between users For people with visual impairments, magnification is not always the most effective strategy (especially with obstructed visual fields).

- Program manager to read and manipulate windows, menus, and icons
- Word processing-based tasks such as opening and saving files, reading and editing text, text attributes, and toolbars
- Spreadsheet tasks such as reading cells, tables figures, and editing data and formulas
- Internet use, including dialing up, accessing World Wide Web pages, navigating with link buttons, sending e-mail and reading graphics
- Screen searching, such as searching for characters, strings, formats, and icons
- Operating start menu, exploring and controlling settings (Becker and Lundman 1998)

For the assessment of seven magnification programs (synthetic speech and Braille displays are not relevant to individuals with visual impairments who have residual vision), the evaluators comprised of a system engineer, ergonomic engineer, computer science expert, and three individuals with visual impairments.

The use of magnification, as a strategy to afford access to GUIs proved somewhat successful, providing 89% or higher access to GUIs, except for Internet use (84% access) and screen searching (0%) (Fortuin and Omata 2004). It was concluded that the essential problem for the design of interactive systems for users with visual impairments is (1) to determine what the users need and (2) how to represent the requested information based on key psychological and physical attributes of the user. The result of ineffective assistive technologies is a lack of usable contextual cues for the users to provide feedback in the case of errors; and this translates to large amounts of imposed workload on the user and frustration (Fortuin and Omata 2004).

In a case study on an English teacher who was having difficulty reading student papers, typing, and proofreading, Whittaker (1998) discovered that magnification was not affording optimal performance. Typically, the authors found that users with visual acuity of 20/40 or better would respond well to simple optical magnification. The authors investigated other visual functioning to find that the individuals had severely diminished contrast sensitivity (13% contrast threshold, with 2% representing normal sensitivity). Furthermore, this individual's visual field was 20 degrees horizontally (180 degrees is normal). Magnification was likely reducing the number of letters viewable simultaneously in the presence of scotoma within the visual field. The author warned that magnifiers and large monitors are not always the most effective solution for users with impaired vision.

Arditi (2004) addressed the reading difficulties of individuals with low vision. According to the author, successfully overcoming this difficulty is accomplished through the exploitation of remaining vision. The easiest way to do this is through magnification, but as shown in this study, it is not a one size fits all solution. Several parameters of the font, including height, stroke, spacing, and serif size, must be selected in a combination that best suits a given user. Arditi presented the prototype and initial user testing of computer-based software that lets a user customize fonts for maximized legibility. Those users studied were able to adjust font to a usable, legible level, to positively impact reading times and the reading acuity. Arditi and Lu (2008) successfully created a novel web browser front end that goes beyond simple magnification and postconfiguration, no additional adjustments are needed, regardless of document type.

An in-depth review of accessibility tools aimed at improving interactions of computer users with low vision informed the design of MouseLupe (Silva and Bellon 2002). MouseLupe simulates a magnifying glass, enabling users to magnify select portions of text or display graphics, inspired by the problematic nature of screen magnification software. The authors suggested that magnification improves the readability of smaller text, but occludes the visible area of the document. Furthermore, graphics that contain text (like most icons), a critical element of the GUI, when enlarged, are difficult to read (for a comprehensive review of magnification tools, see Silva and Bellon [2002]).

Several researchers have considered the effect of visual impairments on web browsing (Arditi 2003; Craven 2003; Harper, Goble, and Stevens 2001; Murphy et al. 2008; Silva and Bellon 2002). Harper, Goble, and Stevens (2001) addressed this problem in terms of "web mobility." These authors provided guidelines for movement through and around complex hypermedia environments, such as the web, for users with visual impairments. The problem, according to these authors, is that visual impairment inhibits individuals' ability to efficiently assimilate page structure and visual cues that lead to the following problems:

- Failure to get a feel for the content on the website
- Failure to have a sense for the magnitude of the display or where in a website the interaction takes place
- Disorientation
- Obstacles and distracters such as spacer images, tables, and large images
- Too much complex detail that cannot be resolved
- Frustration

Arditi (2003) observed the problems of web browsing in terms of the allocation of screen space resources. According to the author, conflicts arise in the implementation of web browsing solutions for individuals with low vision including (1) high magnification requirements; (2) variable typography color, size, and contrasts of the content presented; (3) embedded text messages to augment web images; and (4) accessible web browsing controls (icons, buttons, menus). The author presented a novel approach for effectively using screen resources, providing evidence that the strategic layout of a display is a critical factor to successful interaction. The layout of screen elements was interpreted as more critical than magnification of the screen elements.

Craven (2003) questioned the accessibility of electronic library resources on the World Wide Web for individuals with visual impairments. The results of her study with 20 sighted and 20 visually impaired users revealed the browsing times of those individuals with visual impairments were significantly greater, depending on the design of the website (layout complexity and distracters). Navigation time for the group of users with visual impairments was significantly longer due to visual functioning, but also due to artifacts of assistive technology use in navigation (magnification and screen readers).

Not only are the tools inadequate for many visually impaired users, but many Web pages do not follow accessibility guidelines set forth by the W3C, the World Wide Web Consortium (http://www.w3.org/wai) or the web accessibility guidelines from the U.S. Federal Section 508 (http://www.section508.gov). The two most popular screen readers are JAWS (Job Access with Speech) and Window-Eyes. Despite their popularity, users have identified issues with their accessibility and usability. In a recent survey of visually impaired users, Lazar et al. (2010) found the following as top causes of user frustration to be

- Page layout causing confusing screen reader feedback
- Conflict between the screen reader and application
- Poorly designed/unlabeled form
- No alt text for pictures
- Misleading links
- Inaccessible PDF
- Screen reader crash

Web 2.0 is causing additional challenges for visually impaired web browsing. Applications supported by Web 2.0 are often visually-dynamic, interactive and capable of replicating desktop functionality. A result, stand-alone screen readers are not able to handle many types of Web 2.0 applications. While additional research is needed in this area, many technologies are rapidly evolving to meet this need. Specifications such as Accessible Rich Internet Applications or WAI-ARIA are currently being developed to provide more semantic information about web components and enable greater accessibility. WAI-ARIA has been used for improved accessibility of Wikipedia, a Web 2.0 collaborative encyclopedia that allows users to edit pages (Senette et al. 2009).

Another web solution has been developed by IBM for the visually impaired. World Wide Telecom Web (WWTW), or the Spoken Web, can be used not only for delivering information and services to those with low literacy and technical skills, but also to the visually impaired. IBM has developed a network of VoiceSites that can be created and accessed by voice interaction over an ordinary telephone. Over 10,000 people worldwide are currently using WWTW (Simonite 2011). IBM research reports that through their usability studies they have found the learning curve to be low, and no extensive training is required (Rajput et al. 2008). The authors suggest that WWTW could be a solution to bring social networking sites to the visually impaired.

38.3.3 Summary

Through technological advances in HCI research, the concepts of perceptual and adaptive interfaces have emerged. These two categories of technologies provide vast opportunities for individuals with perceptual impairments to fully access electronic information. More specifically, multimedia and multimodal systems have been shown to accommodate users of varying abilities. However, while these are the most promising approaches for this population, there still exists a chasm between research and practice. Unfortunately, it is rare to see the knowledge generated through HCI research actually implemented into commercially available products. Much work still needs to be done, both in terms of advancing knowledge—through empirical HCI research—and in terms of increasing awareness. Translational efforts need to be made to help disseminate this new knowledge into the design of technologies and devices, so that individuals with perceptual (and other forms of) impairment can better leverage the increasingly ubiquitous technological tools used in everyday life. An evidence-based need for improved design of technology—especially for individuals with sensory of perceptual impairments—is needed to help ensure that the new knowledge continually generated through research is ultimately used to improve human interaction with technology.

38.4 CONCLUSIONS

To provide a context for the topic of perceptual impairments and computing technologies, Figure 38.1 demonstrated that, in order to achieve universal access across classes of computing technologies, *researchers* must be prepared to address the following very challenging issues:

1. Establish empirical links between clinical diagnoses and sets of functional capabilities.
2. Define the set of functional capabilities required to access information technologies.
3. Establish empirical bases for the influence of specific functional capabilities on access to specific classes of technologies.

Furthermore, *designers* need to be empowered to design and create universally designed products that are fully accessible to people who have perceptual impairments. A key element of this is ensuring that designers can actually make use of the universal design resources that are provided to them during the design process (Law et al. 2008a,b).

This chapter aimed to establish a basis for addressing such issues by examining, specifically, visual impairment, several specific diagnoses, and the resulting functional abilities. Finally, groundbreaking advancements in perceptual interfaces, multimodal interfaces, multimedia interfaces, and adaptive interfaces were discussed, which can be applied across a variety of classes of technology in order to enhance the perceptual experience of people who possess such impairments.

ACKNOWLEDGMENTS

This chapter was made possible through funding provided to the first author by the National Science Foundation, as well as the Intel Corporation.

REFERENCES

Age-Related Eye Disease Study Research Group. 2001. A randomized, placebo-controlled, clinical trial of high-dose supplementation with vitamins C and E, beta carotene, and zinc for age-related macular degeneration and vision loss (AREDS Report No. 8). *Arch ophthalmol* 119:1417–36.

Albacete, P. L., S. K. Chang, and G. Polese. 1994. Iconic language design for people with significant speech and multiple impairments. In *Paper Presented at the First Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '94)*. Marina del Rey, CA.

Applegate, W. B., S. T. Miller, J. T. Elam, J. M. Freeman, T. O. Wood, and T. C. Gettlefinger. 1987. Impact of cataract surgery with lens imiplantation on vision and physical function in elderly patients. *Jama* 257:1064–66.

Arditi, A. 2002. Web accessibility and low vision. *Aging Vis* 14(2):2–3.

Arditi, A. 2003. Low vision web browsing and allocation of screen space resources. *Invest Ophthalmol Vis Sci* 44(5):2767.

Arditi, A. 2004. Adjustable typography: An approach to enhancing low vision text accessibility. *Ergonomics* 47(5):469–82.

Arditi, A., and J. Lu. 2008. Accessible web browser interface design for users with low vision. *Hum Fact Ergon Soc Annu Meet Proc* 52(6):576–80. http://www.ingentaconnect.com/content/hfes/hfproc/2008/00000052/00000006/art00014 (accessed March 19, 2011).

Assistive Technologies for Independent Aging: Opportunities and Challenges. 2004. In *Hearing Before the Special Committee on Aging, 108th Cong., 2nd Sess.*, Vol. 145 (Statement of the American Foundation for the Blind).

Baca, J. 1998. Comparing effects of navigational interface modalities on speaker prosodics. In *Paper Presented at the Third Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '98)*. Marina del Rey, CA.

Barreto, A. B., J. A. Jacko, and P. Hugh. 2007. Impact of spatial auditory feedback on the efficiency of iconic human–computer interfaces under conditions of visual impairment. *Comput Human Behav* 23(3):1211–31. doi:DOI: 10.1016/j.chb.2004.12.001

Becker, S., and D. Lundman. 1998. Improving access to computers for blind and visually impaired persons: The development of a test method for usability. In *Paper Presented at the Telemeatics for the Integration of Disabled and Elderly People (TIDE) 1998 Conference*. Helsinki, Finland.

Blattner, M., D. Sumikawa, and R. Greenberg. 1989. Earcons and icons: Their structure and common design principles. *Hum Comput Interact* 4(1):11–44.

Brenner, M. H., B. Curbow, J. C. Javitt, M. W. Legro, and A. Sommer. 1993. Vision change and quality of life in the elderly: Response to cataract surgery and treatment of other chronic ocular conditions. *Arch Ophthalmol* 111(5):680–5.

Bressler, S., B. Munoz, S. Solomon, S. West, and Eye Evaluation (SEE) Study Team. 2008. Racial differences in the prevalence of age-related macular degeneration: The Salisbury Eye Evaluation (SEE) project. *Arch Opthamol* 126(2):241–5. http://www.ncbi.nlm.nih.gov/pubmed/18268216 (accessed March 20, 2011).

Brewster, S. A., P. C. Wright, and A. D. N. Edwards. 1994. The design and evaluation of an auditory-enhanced scrollbar. In *Paper Presented at the ACM Conference on Human Factors in Computing Systems (CHI)*. Boston, MA.

Bühler, C., H. Heck, and R. Wallbruch. 1994. A uniform control interface for various electronic aids. In *Paper Presented at the 4th International Conference on Computers for Handicapped Persons*. Vienna, Austria.

Chang, S.-K., ed. 1990. *Principles of Visual Programming Systems*. Upper Saddle River, NJ: Prentice-Hall, Inc.

Chignell, M., and J. Waterworth. 1997. Multimedia. In *Handbook of Human Factors and Ergonomic*, ed. G. Salvendy, 1808–61. New York, NY: John Wiley & Sons.

Coeckelbergh, T. R. M., F. W. Cornelissen, W. H. Brouwer, and A. C. Kooijam. 2002. The effect of visual field defects on eye movement and practical fitness to drive. *Vision Res* 42:669–77.

Colenbrander, A. 1977. Dimensions of visual performance. *Trans Am Acad Ophthalmol Otolaryngol* 83(2):332–7.

Craven, J. 2003. Access to electronic resources by visually impaired people. *Inf Res* 8(4). Paper no. 156. http://informationr.net/ir/8–4/paper156.html. Accessed March 1, 2011.

Crispien, K., W. Würz, and G. Weber. 1994. Using spatial audio for the enhanced presentation of synthesised speech within screen-readers for blind computer users. In *Paper Presented at the 4th International Conference on Computers for Handicapped Persons*. Vienna, Austria.

Cunningham, C., and N. Coombs. 1997. *Information Access and Adaptive Technology*. Phoenix, AZ: American Council on Education and The Oryx Press.

Danis, C., and J. Karat. 1995. Technology-driven design of speech recognition systems. In *Paper Presented at the Conference on Designing Interactive Systems: Processes, Practices, Methods, & Techniques*. Ann Arbor, MI.

Darvishi, A., V. Guggiana, E. Munteanu, H. Schauer, M. Motavalli, and M. Rauterberg. 1994. Synthesizing nonspeech sound to support blind and visually impaired computer users. In *Paper Presented at the ICCHP*. Vienna, Austria.

Darvishi, A., E. Munteanu, V. Guggiana, H. Schauer, M. Motavalli, and M. Rauterberg. 1994. Automatic impact sound generation for using in nonvisual interfaces. In *Paper Presented at the First Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '94)*. Marina del Rey, CA.

De Bra, P., G.-J. Houben, and H. Wu. 1999. AHAM: A Dexter-based -reference model for adaptive hypermedia. In *Paper Presented at the tenth ACM Conference on Hypertext and Hypermedia: Returning to our Diverse Roots: Returning to our Diverse Roots*. Darmstadt, Germany.

Diabetic Retinopathy Study Research Group. 1979. Four risk factors for severe visual loss in diabetic retinopathy: DRS Report 3. *Arch Ophthalmol* 97:654–5.

Diabetic Retinopathy Study Research Group. 1981. Photocoagulation treatment of proliferative diabetic retinopathy: Clinical application of diabetic retinopathy study (DRS) findings (DRS Report 8). *Arch Ophthalmol* 88:583–600.

Dix, A., J. Finlay, G. Abowd, and R. Beale. 1998. *Human-Computer Interaction*, 2nd ed. New York, NY: Prentice Hall.

Donderi, D. C., and S. B. Murphy. 1983. Predicting activity and satisfaction following cataract surgery. *J Behav Med* 6:313–28.

Early Treatment Diabetic Retinopathy Study Research Group. 1995. Focal photocoagulation treatment of diabetic macular edema (ETDRS Report 19). *Arch Ophthalmol* 113:1144–55.

Edwards, P. J., L. Barnard, V. K. Emery, J. Yi, K. P. Moloney, T. Kongnakorn et al. 2004. Strategic design for users with diabetic retinopathy: Factors influencing performance in a menu-selection task. In *Proceedings of the Sixth International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS ' 04)*, 118–25. Atlanta, GA. New York: ACM.

Edwards, P. J., L. Barnard, V. K. Emery, J. Yi, K. P. Moloney, T. Kongnakorn et al. 2005. Understanding users with diabetic retinopathy: Factors that affect performance in a menu selection task. *Behav Inf Technol* 24(3):175–86.

Edwards, W. K., and E. D. Mynatt. 1994. An architecture for transforming graphical interfaces. In *Paper Presented at the ACM Conference on User Interface Software and Technology (UIST '94)*. Marina del Rey, CA.

Edwards, W. K., E. D. Mynatt, and K. Stockton. 1994. Providing access to graphical interfaces, not graphical screens. In *Paper Presented at the ACM Conference on Assistive and Enabling Technology (ASSETS, ' 94)*. Marina del Rey, CA.

Emery, V. K., P. J. Edwards, J. A. Jacko, K. P. Moloney, L. Barnard, T. Kongnakorn et al. 2003. Toward achieving universal usability for older adults through multimodal feedback. In *Proceedings of the 2003 ACM Conference on Universal Usability*, 46–53. Vancouver, BC. New York: ACM.

Emiliani, P. L. 2001. Special needs and enabling technologies: An evolving approach to accessibility. In *User Interfaces for All—Concepts, Methods and Tools*, ed. C. Stephanidis, 97–113. Mahwah, NJ: Lawrence Erlbaum Associates.

Fletcher, D. C., ed. 1999. *Low Vision Rehabilitation: Caring for the Whole Person*. Hong Kong: Oxford University Press.

Fletcher, D. C., R. Schuchard, C. Livingstone, W. Crane, and S. Hu. 1994. Scanning laser ophthalmoscope macular perimetry and applications for low vision rehabilitation clinicians. *Ophthalmol Clin North Am* 7:257–65.

Fortuin, F. T., and S. Omata. 2004. Designing universal user interfaces—The application of universal design rules to eliminate information barriers for the visually impaired and the elderly. http://www.visionconnection.org (accessed May 19, 2004).

Fraser, J., and C. Gutwin. 2000. A framework of assistive pointers for low vision users. In *Paper Presented at the ACM Conference on Assistive Technologies*. Arlington, VA.

Fricke, J., and H. Baehring. 1994. Displaying laterally moving tactile information. In *Paper Presented at the 4th International Conference on Computers for Handicapped Persons (ICCHP '94)*. Vienna, Austria.

Friedlander, C. 1997. Speech facilities for the reading disabled. *Commun ACM* 40(3):24–5.

Gaver, W. 1989. The SonicFinder: An interface that uses auditory icons. *Hum Comput Interact* 4(1):67–94.

Gerber, E., and C. Kirchner. 2001. Who's surfing? Internet access and computer use by visually impaired youths and adults. *J Vis Impair Blind* 95(3):176–81.

Gillan, D. J. 1998. The psychology of multimedia: Principles of perception and cognition. In *Paper Presented at the 1998 ACM Conference on Human Factors in Computing Systems (CHI '98)*. Los Angeles, CA.

Gragoudas, E. S., A. P. Adamis, E. T. Cunningham Jr., M. Feinsod, and D. R. Guyer. 2004. Pegaptanib for neovascular age-related macular degeneration. *New England Journal of Medicine* 351:2805–16.

Harper, S., C. Goble, and R. Stevens. 2001. Web mobility guidelines for visually impaired surfers. *J Res Pract Inf Technol* 33(1):30–41.

Huang, X., A. Acero, J. Adcock, H.-W. Hon, J. Goldsmith, J. Liu et al. 1996. Whistler: A trainable text-to-speech system. In *Paper Presented at the Fourth International Conference of Spoken Language Processing (ICSLP '96)*. Philadelphia, PA.

Huang, X., A. Acero, F. Alleva, M.-Y. Hwang, L. Jiang, and M. Mahajan. 1995. Microsoft Windows highly intelligent speech recognizer: Whisper. In *Paper Presented at the International Conference on Acoustics, Speech, and Signal Processing*. Detroit, MI.

Ismail, N., and H. Zaman. 2010. Search engine module in voice recognition browser to facilitate the visually impaired in virtual learning (MGSYS VISI-VL). *World Acad Sci Eng Technol* 71:606–10.

Jacko, J. A. 1999. The importance of clinical diagnoses in the prediction of performance on computer-based tasks for low vision users. In *Proceedings of the 9th International Conference on Human-Computer Interaction*, 787–91. Munich.

Jacko, J. A., L. Barnard, T. Kongnakorn, K. P. Moloney, P. J. Edwards, V. K. Emery et al. 2004. Isolating the effects of visual impairment: Exporing the effect of AMD on the utility of multimodal feedback. *CHI Lett* 6(1):311–18.

Jacko, J. A., A. B. Barreto, I. U. Scott, J. Y. M. Chu, H. S. Vitense, F. T. Conway et al. 2002. Macular degeneration and visual icon use: Deriving guidelines for improved access. *Univers Access Inf Soc* 1:197–296.

Jacko, J. A., A. B. Barreto, I. U. Scott, R. H. Rosa Jr., and C. J. Pappas. 2000. Using electroencephalogram to investigate stages of visual search in visually impaired computer users: Preattention and focal attention. *Int J Hum Comput Interact* 12(1):135–50.

Jacko, J. A., M. A. Dixon, R. H. Rosa, I. U. Scott, and C. J. Pappas. 1999a. Linking visual capabilities of partially sighted computer users to psychomotor task performance. In *Proceedings of the 9th International Conference on Human-Computer Interaction*, 975–9. Munich, Germany, August 22–27. New York: ACM.

Jacko, J. A., M. A. Dixon, R. H. Rosa Jr., I. U. Scott, and C. J. Pappas. 1999b. Visual profiles: A critical component of universal access. In *ACM Conference on Human Factors in Computing Systems (CHI '99)*, 330–7. Pittsburg, PA. New York: ACM.

Jacko, J. A., V. K. Emery, P. J. Edwards, M. Ashok, L. Barnard, T. Kongnakorn et al. 2004. The effects of multimodal feedback on older adults' task performance given varying levels of computer experience. *Behav Inf Technol* 23(4):247–64.

Jacko, J. A., K. P. Moloney, T. Kongnakorn, L. Barnard, P. J. Edwards, V. K. Emery et al. 2005. Multimodal feedback as a solution to ocular disease-based user performance decrements in the absence of functional visual loss. *Int J Hum Comput Interact* 18(2):183–218.

Jacko, J. A., R. H. Rosa, I. U. Scott, C. J. Pappas, and M. A. Dixon. 2000. Visual impairment: The use of visual profiles in evaluations of icon use in computer-based tasks. *Int J Hum Comput Interact* 12(1):151–65.

Jacko, J. A., I. U. Scott, A. B. Barreto, H. S. Bautsch, J. Y. M. Chu, and W. B. Fain. 2001. Iconic visual search strategies: A comparison of computer users with AMD versus computer users with normal vision. In *Paper Presented at the 9th International Conference on Human-Computer Interaction*. New Orleans, LA.

Jacko, J. A., I. U. Scott, F. Sainfort, K. P. Moloney, T. Kongnakorn, B. S. Zorich et al. 2003. Effects of multimodal feedback on the performance of older adults with normal and impaired vision. *Lect Notes Comput Sci (LNCS)* 2615:3–22.

Jacko, J. A., and A. Sears. 1998. Designing interfaces for an overlooked user group: Considering the visual profiles of partially sighted users. In *Paper Presented at the ACM Conference on Assistive Technologies*. Marina del Rey, CA.

Jacko, J. A., and H. S. Vitense. 2001. A review and reappraisal of information technologies within a conceptual framework for individuals with disabilities. *Univers Access Inf Soc (UAIS)* 1:56–76.

Jacko, J. A., H. S. Vitense, and I. U. Scott. 2003. Perceptual impairments and computing technologies. In *Human-Computer Interaction Handbook*, ed. J. A. Jacko and A. Sears, 504–22. Mahwah, NJ: Lawrence Erlbaum Associates.

Javitt, J. C., M. H. Brenner, B. Curbow, M. W. Legro, and D. A. Street. 1993. Outcomes of cataract surgery. Improvement in visual acuity and subjective visual function after surgery in the first, second, and both eyes. *Arch Ophthalmol* 111(5):686–91.

Karshmer, A. I., P. Brawner, and G. Reiswig. 1994. An experimental sound-based hierarchical menu navigation system for visually handicapped use of graphical user interfaces. In *Proceedings of the First Annual ACM Conference on Assistive Technologies*, 123–8. Marina del Rey, CA. New York: ACM.

Karshmer, A. I., B. Ogden, P. Brawner, K. Kaugars, and G. Reiswig. 1994. Adapting graphical user interfaces for use by visually handicapped computer users: Current results and continuing research. In *Paper Presented at the 4th International Conference on Computers for Handicapped Persons (ICCHP '94)*. Vienna, Austria.

Kawai, S., H. Aida, and T. Saito. 1996. Designing interface toolkit with dynamic selectable modality. In *Paper Presented at the Second Annual ACM Conference on Assistive Technologies*. Vancouver, Canada.

Klein, R., B. Klein, M. Knudtsen, T. Wong, M. Cotch, K. Liu et al. 2006. Prevalence of age-related macular degeneration in 4 racial/ethnic groups in the multi-ethnic study of atherosclerosis. *Opthamology* 113(3):373–80. http://www.ncbi.nlm.nih.gov/pubmed/16513455 (accessed March 20, 2011).

Kline, R. L., and E. P. Glinert. 1995. Improving GUI accessibility for people with low vision. In *Paper Presented at the SIGCHI Conference on Human Factors in Computing Systems*. Denver, CO.

Kline, D. W., and C. T. Scialfa. 1997. Sensory and perceptual functioning: Basic research and human factor implications. In *Handbook of Human Factors and the Older Adult*, ed. A. D. Fisk and W. A. Rogers, 27–54. San Diego, CA: Academic Press.

Kuber, R., W. Yu, and M. S. O'Modhrain. 2010. Tactile web browsing for blind users. In *Haptic and Audio Interaction Design: 5th International Workshop, HAID 2010, Copenhagen, Denmark, September 16–17, 2010; Proceedings*, ed. R. Nordahl, S. Serafin, F. Fontana, and S. Brewster, Vol. 6306/2010, 75–84. Berlin: Springer. Print. Lecture Notes in Computer Science.

Kurze, M. 1998. TGuide: A guidance system for tactile image exploration. In *Paper Presented at the Third Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '98)*. Marina Del Rey, CA.

LaPlante, M. P. 1988. Prevalence of conditions causing need for assistance in activities of daily living. In *Data on disability from the National Health Interview Survey, 1983–1985*, ed. M. P. LaPlante, 3. Washington, DC: National Institute on Disability and Rehabilitation Research.

Law, C. M., J. S. Yi, Y. S. Choi, and J. A. Jacko. 2008a. A systematic examination of universal design resources: Part 1, heuristic evaluation. *Univers Access Inf Soc* 7(1–2):31–54.

Law, C. M., J. S. Yi, Y. S. Choi, and J. A. Jacko. 2008b. A systematic examination of universal design resources: Part 2, analysis of the development process. *Univers Access Inf Soc* 7(1–2):55–77.

Lazar, J., A. Allen, J. Kleinman, and C. Malarkey. 2010. What frustrates screen reader users on the web: A study of 100 blind users. *Int J Hum Comput Interact* 22(3):247–69. http://triton.towson.edu/~jlazar/IJHCI_blind_user_frustration.pdf (accessed March 21, 2011).

Lee, H.-W., G. E. Legge, and A. Oritz. 2003. Is word recognition different in central and peripheral vision? *Vision Res* 43:2837–46.

Leonard, V. K., P. J. Edwards, and J. A. Jacko. 2005. Informing accessible design through self-reported quality of visual health. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*, 994–8. Orlando, FL. New York: ACM.

Leonard, V. K., J. A. Jacko, and J. J. Pizzimenti. 2005. An exploratory investigation of handheld computer interaction for older adults with visual impairments. In *Proceedings of the 7th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2005)*, 12–9. Baltimore, MD. New York: ACM.

McCoy, K. F., P. Demasco, C. A. Pennington, and A. L. Badman. 1997. Some interface issues in developing intelligent communication aids for people with disabilities. In *Paper Presented at the 1997 International Conference on Intelligent User Interfaces (IUI '97)*. Orlando, FL.

McMillan, W. W., and L. Wisniewski. 1994. A rule-based system that suggests computer adaptations for users with special needs. In *Paper Presented at the First Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '94)*. Marina del Rey, CA.

Mehr, E. B., and A. N. Freid. 1975. *Low Vision Care*. Chicago: Professional Press.

Mereu, S. W., and R. Kazman. 1996. Audio enhanced 3D interfaces for visually impaired users. In *Paper Presented at the 1996 ACM Conference on Human Factors in Computing Systems (CHI '96)*. Vancouver, Canada.

Mitzner, T., D. Touron, W. Rogers, and C. Hertzog. 2010. Checking it twice: Age-related differences in double checking during visual search. In *Human Factors and Ergonomics Society Annual Meeting Proceedings, Perception & Performance*, Vol. 5, 1326–30. http://www.ingentaconnect.com/content/hfes/hfproc/2010/00000054/00000018/art00005 (accessed March 19, 2011).

Moloney, K. P., B. Shi, V. K. Leonard, J. A. Jacko, B. Vidakovic, and F. Sainfort. 2006. Leveraging data complexity: Pupillary behavior of older adults with visual impairment during HCI. *Trans Comput Hum Interact (TOCHI)*.

Murphy, E., R. Kuber, G. McAllister, P. Strain, and W. Yu. 2008. An empirical investigation into the difficulties experienced by visually impaired internet users. *Univers Access Inf Soc* 7(1):79–91.

Mynatt, E. D., and G. Weber. 1994. Nonvisual presentation of graphical user interfaces: Contrasting two approaches. In *Paper Presented at the 1994 ACM Conference on Human Factors in Computing Systems (CHI '94)*. Boston, MA.

Nguyen, N. X., M. Weismann, and S. Trauzettel-Klosinski. 2009. Improvement of reading speed after providing of low vision aids in patients with age-related macular degeneration. *Acta Ophthalmologica* 87(8):849–53. doi:10.1111/j.1755–3768.2008.01423.x

Nilsson, U. L. 1990. Visual rehabilitation of patients with and without educational training in the use of optical aids and residual vision: A prospective study of patients with advanced age-related macular degeneration. *Clin Vision Sci* 6:3–10.

Nilsson, U. L., and S. E. G. Nilsson. 1986. Rehabilitation of the visually handicapped with advanced macular degeneration. *Doc Ophthalmol* 62(4):345–67.

Oakley, I., M. R. McGee, S. A. Brewster, and O. Gray. 2000. Putting the feel in 'look and feel'. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Hague*, 415–22. The Netherlands. New York: ACM.

Orr, A. L. 1998. *Issues in Aging and Vision*. New York: American Foundation for the Blind.

Oviatt, S. L. 1999. Mutual disambiguation of recognition errors in a multimodel architecture. In *Paper Presented at the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit (CHI '99)*. Pittsburgh, PA.

Oviatt, S. L., and P. Cohen. 2000. Multimodal interfaces that process what comes naturally. *Commun ACM* 43(3):45–53.

Oviatt, S., Cohen, P., Suhm, B., Bers, J., Wu, I., Holzman, T., Winograd, T., Vergo, J., Duncan, L., Landay, J., Larson, J., and Ferro, D. 2002. Designing the user interface for multimodal speech and gesture applications: state-of-the-art systems and research directions from 2000 and beyond. In *Human-Computer Interaction in the New Millennium*, ed. J. M. Carroll, 419–56. Reading, MA: Addison-Wesley.

Paciello, M. G. 1996. Designing for people with disabilities. *Interactions* 3(1):15–6.

Parrish, R. K. I., S. J. Gedde, I. U. Scott, W. J. Feuer, J. C. Schiffman, C. M. Mangione et al. 1997. Visual function and quality of life among patients with glaucoma. *Arch Ophthalmol* 115(11):1447–55.

Paternó, F., and C. Mancini. 1998. Developing adaptable hypermedia. In *Paper Presented at the 4th International Conference on Intelligent User Interfaces (IUI '99)*. Los Angeles, CA.

Petrie, H., S. Morley, and G. Weber. 1995. Tactile-based direct manipulation in GUIs for blind users. In *Paper Presented at the Conference Companion on Human Factors in Computing Systems (CHI '95)*. Denver, CO.

Pilar da Silva, D., R. Van Durm, E. Duval, and H. Olivië. 1998. Adaptive navigational facilities in educational hypermedia. In *Paper Presented at the Ninth ACM Conference on Hypertext and Hypermedia: Links, Objects, Time and Space—Structure in Hypermedia Systems (HyperText '98)*. Pittsburgh, PA.

Rajput, N., S. Agarwal, A. Kumar, and A. A. Nanavati. 2008. An alternative information web for visually impaired users in developing countries. *IBM Research Report*, *RI08010*, 1–9. http://domino.research.ibm.com/library/cyberdig.nsf/papers/E1642C08F06FA89C85257491003E16D0/$File/IBMresearchReport.pdf (accessed March 20, 2011).

Rampp, D. L. 1979. Hearing and learning disabilities. In *Hearing and Hearing Impairment*, ed. L. J. Bradford and W. G. Hardy, 381–9. New York: Grune & Stratton.

Ramstein, C. 1996. Combining hepatic and Braille technologies: Design issues and pilot study. In *Paper Presented at the Second Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '96)*. Vancouver, Canada.

Reeves, B., and C. Nass. 2000. Perceptual user interfaces: Perceptual bandwidth. *Commun ACM* 43(3):65–70.

Rosenberg, R., E. Faye, M. Fischer, and D. Budicks. 1989. Role of prism relocation in improving visual performance of patients with macular dysfunction. *Optom Vis Sci* 66(11):747–50.

Rovner, B. W., R. J. Casten, M. T. Hegel, R. W. Massof, B. E. Leiby, and W. S. Tasman. 2011. Improving function in age-related macular degeneration: Design and methods of a randomized clinical trial. *Contemp Clin Trials* 32(2):196–203. doi:10.1016/j.cct.2010.10.008

Roy, D., and A. Pentland. 1998. A phoneme probability display for individuals with hearing disabilities. In *Paper Presented at the Third Annual International ACM/SIGCAPH Conference on Assistive Technologies (ASSETS '98)*. Marina del Rey, CA.

Ryder, J. W., and K. Ghose. 1999. Multi-sensory browser and editor model. In *Paper Presented at the 1999 ACM Symposium on Applied Computing*. San Antonio, TX.

Schank, R. C. 1972. Conceptual dependency: A theory of natural language understanding. *Cogn Psychol* 3(4):532–631.

Schieber, F. 1994. *Recent Developments in Vision, Aging and Driving: 1988–1994*. Report No. UMTRI-94-26. Ann Arbor, MI: University of Michigan, Transportaion Research Institute.

Schumacher, E. H., J. A. Jacko, S. A. Primo, K. L. Main, K. P. Moloney, E. N. Kinzel, J. Ginn. 2008. Reorganization of visual processing is related to eccentric viewing in patients with Macular Degeneration. *Restor Neuro Neurosci* 26(4–5):391–402.

Scott, I. U., W. J. Feuer, and J. A. Jacko. 2002a. Impact of graphic user interface screen features on computer task accuracy and speed in a cohort of patients with age-related macular degeneration. *Am J Ophthalmol* 134(6):857–62.

Scott, I. U., W. J. Feuer, and J. A. Jacko. 2002b. Impact of visual function on computer task accuracy and reaction time in a cohort of patients with age-related macular degeneration. *Am J Ophthalmol* 133(3):350–7.

Scott, I. U., J. A. Jacko, F. Sainfort, V. K. Leonard, T. Kongnakorn, and K. P. Moloney. 2006. The impact of auditory and haptic feedback on computer task performance in patients with Age-related Macular Degeneration and control subjects with no known ocular disease. *Retina* 26(7):803–10.

Scott, I. U., O. D. Schein, W. J. Feuer, M. F. Folstein, and K. Bandeen-Roche. 2001. Emotional distress in patients with retinal disease. *Am J Ophthalmol* 131(5):584–9.

Scott, I. U., O. D. Schein, S. West, K. Bandeen-Roche, C. Enger, and M. F. Folstein. 1994. Functional status and quality of life measurement among ophthalmic patients. *Arch Ophthalmol* 112(3):329–35.

Scott, I. U., W. E. Smiddy, W. Feuer, and A. Merikansky. 1998. Vitreoretinal surgery outcomes: Results of a patient satisfaction/functional status survey. *Ophthalmology* 105(5):795–803.

Scott, I. U., W. E. Smiddy, J. Schiffman, W. J. Feuer, and C. J. Pappas. 1999. Quality of life of low-vision patients and the impact of low-vision services. *Am J Ophthalmol* 128(1):54–62.

Senette, C., M. Buzzi, M. Buzzi, and B. Leporini. 2009. Enhancing wikipedia editing with WAI-ARIA. In *HCI and Usability for e-Inclusion, Volume 5889 of Lecture Notes in Computer Science*, ed. A. Holzinger and K. Miesenberger. Chapter 11, 159–77. Berlin, Heidelberg: Springer Berlin/Heidelberg.

Silva, L., and O. R. P. Bellon. 2002. A novel application to aid low vision computer users. *Lect Notes Comput Sci* 2398:455–62.

Simonite, T. 2011. A world wide web that talks - technology review. *Technology Review: The Authority on the Future of Technology*. http://www.technologyreview.com/printer_friendly_article.aspx?id (accessed March 21, 2011).

Sloan, L. L. 1968. Reading aids for the partially sighted: Factors which determine success or failure. *Arch Ophthalmol* 80(1):35–8.

Steinberg, E. P., J. M. Tielsch, O. D. Schein, J. C. Javitt, P. Sharkey, S. D. Cassard et al. 1994. National study of cataract surgery outcomes: Variation in 4-month postoperative outcomes as reflected in multiple outcome measures. *Ophthalmology* 101(6):1131–40.

Stelmack, J. A., X. C. Tang, D. J. Reda, S. Rinne, R. M. Mancil, R. W. Massof, and for the LOVIT Study Group. 2008. Outcomes of the veterans affairs low vision intervention trial (LOVIT). *Arch Ophthalmol* 126(5):608–17. doi:10.1001/archopht.126.5.608

Stephanidis, C. 2001a. The concept of unified user interfaces. In *User Interfaces for All: Concepts, Methods, and Tools*, ed. C. Stephanidis, 371–88. Mahwah, NJ: Lawrence Erlbaum Associates.

Stephanidis, C., ed. 2001b. *User Interfaces for All: Concepts, Methods, and Tools*. Mahwah, NJ: Lawrence Erlbaum Associates.

Stephanidis, C., C. Karagiannidis, and A. Koumpis. 1997. Decision making in intelligent user interfaces. In *Paper Presented at the 2nd International Conference on Intelligent User Interfaces (IUI '97)*. Orlando, FL.

Stephanidis, C., G. Salvendy, D. Akoumianakis, A. Arnold, N. Bevan, D. Dardailler et al. 1999. Toward an information society for all: HCI challenges and R&D recommendations. *Int J Hum Comput Interact* 11(1):1–28.

Stephanidis, C., G. Salvendy, D. Akoumianakis, N. Bevan, J. Brewer, P. L. Emiliani et al. 1998. Toward an information society for all: An international research and development agenda. *Int J Hum Comput Interact* 10(2):107–34.

Thatcher, J. 1994. Screen reader/2: Access to OS/2 and the graphical user interface. In *Paper Presented at the First Annual ACM Conference on Assistive Technologies (ASSETS '94)*. Marina del Rey, CA.

The Eye Diseases Prevalence Research Group. 2004a. Causes and prevalence of visual impairment among adults in the United States. *Arch Ophthalmol* 122:477–85.

The Eye Diseases Prevalence Research Group. 2004b. Prevalence of age-related macular degeneration in the United States. *Arch Ophthalmol* 122:564–72.

The Eye Diseases Prevalence Research Group. 2004c. The prevalence of diabetic retinopathy among adults in the United States. *Arch Ophthalmol* 122:552–63.

Turk, M., and G. Robertson. 2000. Perceptual user interfaces. *Commun ACM* 43(3):32–4.

Vanderheiden, G. C. 1998. Universal design and assistive technology in communication and information technologies: Alternatives or complements? *Assistive Technology* 10(1):29–36.

Vitense, H. S., J. A. Jacko, and V. K. Emery. 2002. Foundations for improved interaction by individuals with visual impairments through multimodal feedback. *Univers Access Inf Soc (UAIS)* 2(1):76–87.

Vitense, H. S., J. A. Jacko, and V. K. Emery. 2003. Multimodal feedback: An assessment of performance and mental workload. *Ergonomics* 46(1–3):58–87.

Waterworth, J. A., and M. H. Chignell. 1997. Multimedia interaction. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. K. Landauer, and P. Prabhu, 915–46. New York, NY: Elsevier Science.

Whittaker, S. G. 1998. Choosing assistive devices when computer users have impaired vision. In *Paper Presented at the Center on Disabilities, Technology, and Persons with Disabilities Conference*. Northridge, CA.

Yankelovich, N., G.-A. Levow, and M. Marx. 1995. Designing SpeechActs: Issues in speech user interfaces. In *Paper Presented at the SIGCHI Conference on Human Factors in Computing Systems*. Denver, CO.

Yu, W., R. Ramloll, and S. Brewster. 2000. Haptic graphs for blind computer users. In *Paper Presented at the First Workshop on Haptic Human-Computer Interaction*. Glasgow, Scotland.

39 Universal Accessibility and Low-Literacy Populations

Implications for Human–Computer Interaction Design and Research Methods

William M. Gribbons

CONTENTS

39.1 INTRODUCTION

Over the past decade, the topic of universal accessibility has received close attention in the field of system design. Although considerable progress has been made in addressing the needs of the physically disabled and aging, low-literacy populations have largely been overlooked (Gribbons 1992; Newell et al. 2003; Lewis 2006; Friedman and Bryen 2007). With nearly 45 million adult Americans suffering from the debilitating effects of illiteracy, the universal accessibility movement has clearly not been universal. While many have called for action (Dickinson, Eisma, and Gregor 2003; Shneiderman 2000; Shneiderman 2007), an extensive review of the leading human factors and human–computer interaction (HCI) journals revealed a small number of studies focused on this critical issue. And while the number of entries on this topic in the hcibib (http://hcibib.org/) has grown over the past 5 years, this population is grossly underrepresented compared with other disabilities. Most significantly, the topic of low literacy, learning disabilities, and cognitively disabilities remain glaringly absent from the leading HCI and human factors textbooks.

The research that does exist in low literacy and learning disability studies is concentrated, to a large degree, in the fields of educational psychology, instructional design, and health care informatics. Here, a rich research tradition dates back many decades. However, many of these findings are limited to the design of the classroom experience, standalone computer-based training systems, patient interventions, and paper-based communication products. Research in non-educational settings for general interface and interaction design is limited. Compounding the problem, much of this work focuses on children rather than adults. Over the past 5 years, some progress, as noted in this chapter, has been made. Given the size of this population, it is also interesting to note that the fields of marketing and consumer behavior have directed considerable attention to accommodating the unique requirements of this group (Adkins and Ozanne 2005; Ozanne, Adkins, and Sandlin 2005; Viswanathan, Rosa, and Harris 2005). Similarly, the library sciences has considered the needs of this population while considering the accessibility of online library databases (Stewart, Narendra, and Schmetzke 2005; Craven and Booth 2006). And finally, the special needs of this population have been considered in the design of public transportation systems (Carmien et al. 2005).

The ubiquitous nature of the Internet, expansion in the use of information and communication technologies (ICT), e-health and e-government, and consumer electronics has motivated recent progress on accommodations for low literacy and related populations. The Web Content Accessibility Guidelines (WCAG) 1.0 (1999) released by the World Wide Web Consortium (W3C) urges designers to accommodate this population. Unfortunately, emerging guidelines continue to be vague and nonspecific such as those found at the Center for Usable Design. And just as we begin to make progress in one area, new applications of technologies, such as mobile access to the web in developing nations (Shneiderman 2007; and Sherwani et al. 2009) and the rapid growth of e-health technologies in the home (Bogner 1999), raise new issues for designers and researchers alike.

This chapter will address the challenge of accommodating low-literacy users in system design. It will explore the nature of this population and the magnitude of the problem, the relationship between literacy, learning disability, and cognitive disabilities, general characteristics, functional characteristics, and recommendations for design and research best practice. Ultimately, it is hoped that this effort will encourage further research in this critical area and the integration of that work in our profession's educational programs, guidelines, and best practices.

39.1.1 Defining the Population

Part of the challenge of addressing this population is building consensus on the most effective definition of this group. Common labels include "low literacy," "functionally illiterate," "cognitively disabled," "print-disabled," "learning disabled," (a subset of the cognitively disabled group) or specific learning disabilities such as dyslexia (Figure 39.1). Friedman and Bryen (2007) speculate the lack of agreed upon definition of this population is a contributing factor to the lack of focused research. An underlying assumption in this chapter is that these categories are not mutually exclusive nor the boundaries between each well defined. Because of the lack of clear boundaries, this chapter will focus on functional characteristics rather than clinical definitions. It is also acknowledged that a given disability can range from moderate to severe, as depicted in Figure 39.1.

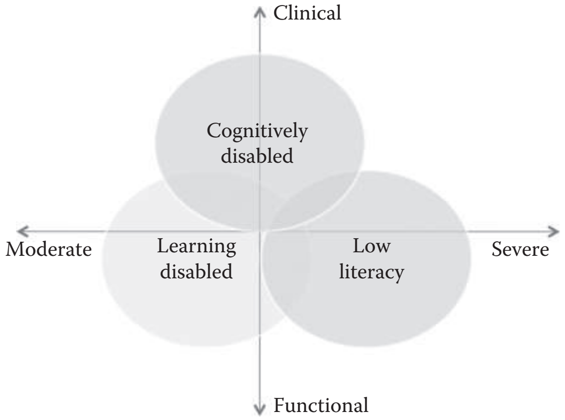


FIGURE 39.1 Comprehensive model of population.

For the purposes of this chapter, discussion will focus on low-literacy and learning-disabled populations and universal design solutions rather than more severe cognitive disabilities requiring assistive technologies. Functional illiteracy, a lack of document and quantitative literacy needed to function in modern society, was selected because it provides the most comprehensive picture of the total population. This perspective brings to the discussion a very detailed set of demographics and statistics. Unfortunately, this perspective is weak on underlying causes and effective accommodations. Fortunately, the learning-disability perspective provides an extensive research base that defines the characteristics of this population, the underlying source of the disability, and possible accommodations. Patterson (2008) summarizes numerous studies associating learning disabilities with low-literacy skills in adults and the persistence of the disability throughout one's life. Learning disability is an umbrella term used to describe a wide range of disorders in information processing and it is generally believed that learning disabilities are linked to a dysfunction in the central nervous system. The complete cognitively disabled category was not chosen since it includes a much broader class of disabilities including Down syndrome, autism, aphasia, emotional disabilities, and Alzheimer's disease. Naturally, these more severe disabilities require more extreme accommodations. However, it is highly likely that some adults identified as low literacy might suffer from an undiagnosed cognitive disability. Consequently, effects of "mild" cognitive disabilities will also be considered. It is estimated that nearly 80% of those diagnosed with a cognitive disability have a mild disability (Carmien et al. 2005).

The causes of low literacy are complex, varied and often inextricably intertwined. Underlying these causes include a lack of educational attainment, social deprivation, learning disabilities, or other cognitive disability. Further complicating matters, many adults with low-literacy skills were never diagnosed with a learning disability as children and entered adulthood suffering the debilitating effects of the disability without knowing the cause. Consequently, it is all but impossible to characterize whether the cause of low literacy in adults is a product of low educational attainment, social deprivation, an underlying disability, or whether an undiagnosed disability was the cause of an individual dropping out of the educational system.

39.1.2 Population Size

The current state of functional illiteracy in the United States paints a shameful picture. According to the 1992 National Adult Literacy Survey, some 23%, or nearly 45 million of approximately 200 million adult Americans, function at the lowest level of literacy, "Level 1." Those at Level 1 literacy are, for the most part, able only to read a simple form or understand rudimentary information in a short news article; others are not capable of even this (Kirsch et al. 1993a). Given the magnitude of these figures, it is surprising that greater attention has not been directed to this population, since the low-literacy population outnumbers all other disability groups combined.

The other literacy levels, two through five, represent progressively higher levels of literacy skills. Not surprisingly, one contributor to functional illiteracy is low educational attainment. The National Adult Literacy Survey states that: "adults with relatively few years of education were more likely to perform in the lower-literacy levels than those who completed high school or received some postsecondary education" (Kirsch et al. 1993a, p. xviii). The National Center for Educational Statistics (2001) reported the following:

> Between 1972 and 1985, high school completion rates climbed by 2.6% points (from 82.8% in 1972 to 85.4% in 1985); since 1985, the rate has shown no consistent trend and has fluctuated between 85% and 87%. This net increase of about 3% points over 29 years represents slow progress toward improving the national high school completion rates. (Kirsch et al. 1993b, p. 25.)

Given the lack of progress in this area, it is highly probable that educational attainment will continue to be a key contributor to lower-literacy skills among adults for the foreseeable future.

In 2003, this survey was re-administered and the results showed no significant improvement in prose and document literacy, with some improvement in quantitative literacy (National Assessment of Adult Literacy). Cleary the data from these two studies show the problem is large and progress is slow.

Unlike the measure provided by the National Adult Literacy Survey, there is no large-scale study measuring the prevalence of learning disabilities in the general population. However, there are a range of estimates from a variety of sources. The National Dyslexia Association (n.d.) estimates that 70%–80% of the population of students receiving special education services has deficits in reading and that 15%–20% of the general population has language-based learning disabilities (how common are language-based learning disabilities simply to document the scope of the problem). Shaywitz (1998) estimates the prevalence of dyslexia among school-age children at 5%–17% of the total population. The Interagency Committee on Learning Disabilities believes that 5%–10% of the population is affected by learning disabilities. The President's Committee on Employment for People with Disabilities found that 10%–14% of the adults in the workforce have learning disabilities. Finally, the National Institutes of Health estimated that 15% of the population in the United States has some type of learning disability (National Institute for Literacy, n.d.). Despite the probable link between those with learning disabilities and the larger lower-literacy population, few attempts have been made to validate this connection.

Not surprisingly, low literacy is not confined to the United States. There are an estimated 876 million illiterate adults in the world, which represents nearly a quarter of the world's population. In developed nations, it is estimated that there are approximately 100 million functionally illiterate people (Kickbusch 2001). In Canada, it is estimated that 30% of adult Canadians have low-literacy skills (Brez and Taylor 1997). In India alone, nearly 45% of the adult population is illiterate according to the 2001 Indian Census (Huenerfauth 2002). These statistics and others clearly convey the global scale of this problem.

39.1.3 Making the Case for Including Functional Illiteracy

The lack of research addressing literacy and learning disability in the HCI discipline was understandable 10 or 15 years ago when this population was unlikely to interact with technology outside of an educational setting. With the ubiquitous nature of information technology in modern society (e.g., the Internet, computers in all aspects of work, public kiosks, ATMs, public transportation, e-banking, e-government, e-health, and consumer electronics), this research gap is now indefensible. Accessible technology is no longer a luxury or an option; rather it is a prerequisite to gaining access to information that affects one's quality of life and health, participation in government, and contribution to the economy. Lewis (2010) cites a Microsoft-commissioned study that estimates that 16% of computer users in the United States have a cognitive disability. A report from the U.S. Department of Commerce (2002, p. 56) stated that: "between December 1998 and September 2001, Internet use by individuals in the lowest-income households (lower socio-economic groups represent a disproportionately high percentage of the lower-literacy population) increased at a 25 percent annual growth rate." The study also reported that: "as of September 2001, about 65 million of the 115 million adults who were employed and 25 or over use a computer at work" (p. 57). Although the percentage of workers using computers was significantly higher for "professional" positions, 20% of those working as operators, fabricators, laborers, farmers, and fishermen use a computer as part of their work (p. 58). Lastly, shifts in the economy from "hands-on work" to "information-and-technology work" will seriously disadvantage lower-literacy groups unless there is a corresponding increase in functional literacy skills.

If the size of this population alone were not enough to mobilize action, the cognitively disabled are a protected group under the Americans with Disabilities Act in the United States, the Disability Discrimination Act in the United Kingdom, Australia's Disability Discrimination Act and similar acts in other countries. In the United States, employers are required to provide workplace accommodations for employees who disclose their disability. That last stipulation of the act could very well be one of the reasons this population is often "invisible." Low literacy carries a stigma for most people and creates a reluctance to disclose the problem and encourages the development of elaborate means of disguising the disability—no disclosure, no requirement for accommodation. Beyond the stigma, Brez and Taylor (1997) report that many low-literacy adults fear disclosure will result in questioning their competency as an adult, parent, or caregiver.

Finally, there is a growing economic incentive for accommodating the needs of the low-literacy user. Viswanathan, Rosa, and Harris (2005) found, in their study of low-literacy consumers, a high degree of brand loyalty toward companies sensitive to their needs. They calculate that this market sector might represent as much as $380 billion in annual spending. We will see additional incentives in Section 39.1.3.1 that follows.

39.1.3.1 Low Literacy and Health Care

Perhaps the most poignant case of a digital divide between the literate haves and the low literacy have-nots is in the health care community. At the most basic level of following written or spoken instructions, low-literacy patients display an extremely low level of understanding of and compliance to instructions. Hussey (1994) reported the problem is particularly acute among older low-literacy patients. Wolf et al. (2006) found in their sample of low-literacy patients rates of correct interpretation of warning labels on prescription medications ranged from 0% to 78%. With the exception of one label, less than half of the subjects were able to adequately interpret the warnings. Although nontechnology based, one can only imagine how these basic information-processing problems are amplified in technology enabled environments. Further, many of the themes underlying the misunderstanding of written labels—multistep instructions, reading level of text, poor icons, poor use of color, and message clarity are common elements in technology-based systems.

Combine these basic information-processing problems with shorter hospital stays, the shift to out-patient care, patient-managed care, and care in the home and the urgency to consider low-literacy users increases dramatically. (Houts et al. 2001; Epstein, Maley, and Suri 2006). Coinciding with these trends is the growing use of fairly sophisticated technologies including infusion pumps, monitoring devices, and dialysis machines in the home. This deployment is more likely to occur in the homes of the elderly—with higher rates of low literacy and diminished cognitive capability—and be operated and maintained by an elderly caregiver or home health aide—workers who are often poorly paid, poorly educated, and not well regulated (Bogner 1999; Stone and Wiener 2001; Henriksen, Joseph, and Zayas-Caban 2009; Epstein, Maley, and Suri 2006). Epstein, Maley, and Suri (2006) warn that the growing use of distributed diagnostics and the expanding home health care industry must address the requirements of patients actively managing their care and a workforce that varies in training and literacy.

After a long delay, the health care community has made progress in using ICT to enhance the quality of patient care, specifically in disseminating health-related information, monitoring patient care, and storing patient information. As a result, the trend for seeking health-related information online has increased dramatically. An estimated 70 million Americans have sought health information online (Cain et al. 2000) from the nearly 10,000 or more health-related websites (Benton Foundation 1999). Birru et al. (2004) estimated that between 40% and 54% of patients use the Internet to search for information on ailments and treatments. This development would be exciting, given the documented contributions of information to improved health, if it were not for the fact that equal access to this information does not exist for all Americans. Similar to the progression that played out in the larger development community, initial design efforts in the health care sector have focused first on the "typical" fully functioning user and then on the aging and physically disabled. Most health care materials—web-based or paper—are written well above the literacy level of the average American, approximately a 10th-grade level or greater. (Doak, Doak, and Root 1996; Birru et al. 2004; Weiner et al. 2004). Kaphingst, Zanfini, and Emmons (2006) examined the accessibility of web sites containing colorectal cancer information and found the average site was written at a grade level of 12.8.

The health care sector has both a moral and economic responsibility to address the needs of lower-literacy populations. Morally, it is only right that all citizens should benefit equally from the value information technology brings to the quality of health care services. Cashen, Dykes, and Gerber (2004) provided support for this position by reporting that literacy is a better predictor of health status than age, income, employment status, education, or race. It is not surprising that low-literacy adults are twice as likely to be hospitalized as their functionally literate counterparts (Birru et al. 2004). There is clearly an economic motivation as well. A conservative estimate places excess health care costs tied to low literacy at $73 billion a year. (Rudd, Moeykens, and Colton 1999).

In recent years, the definition of the digital divide has shifted from economics to literacy. The availability of low-cost equipment and better access to the Internet in public facilities has lessened economic barriers. According to Shneiderman (2000), poor interface and interaction design remain as one of the defining variables between the technology haves and have-nots. And while we might hold out hope that we may one day eradicate low literacy at the source, the fact that learning disabilities are lifelong and that fewer than one in eight low-literacy workers receives literacy training in the workplace (Sum 1999, p. 156) strongly suggests that the burden falls on universal design.

39.2 GENERAL CHARACTERISTICS OF THE POPULATION

Low literacy is often referred to as the "invisible" disability. In many cases, the individual is unaware of the problem; in others, they "hide" the disability. Findings from the National Literacy Survey (Kirsch et al. 1993a,b) suggest that as many as: "66 to 75 percent of the adults in the lowest [literacy] level and 93 to 97 percent in the second lowest level described themselves as being able to read or write English 'well' or 'very well'" (The Literacy Skills of American Adults, p. xvii). In another study by Moon et al. (1998) reported that 70% of the participants reported they read "really well," while in actuality their reading scores reflected a seventh- to eighth-grade ability. Also contributing to the under-reported size of the population is the stigma attached to low literacy that increases an individual's reluctance to disclose the problem. In addition, the many challenges and failures experienced by the learning disabled over a lifetime often result in a poor concept of self-worth and low self-esteem (Gerber 1998).

As previously discussed, a major cause of low literacy is learning disabilities. The learning disability umbrella encompasses a wide range of information-processing disorders, such as dyslexia (language), dyscalculia (mathematics), and dysgraphia (handwriting). The disability is thought to be neurobiological in origin and present itself in various combinations and levels of severity. In addition, an individual with a deficit in one area may have strengths in other areas. Understanding the heterogeneous nature of the learning-disabled population is critical to formulating appropriate accommodations. Learning disabilities generally persist over a lifetime, but the manner in which the disability presents itself will change with life stages as individuals construct increasingly sophisticated coping strategies (Gerber 1998). The lifelong persistence of the disability, combined with the aforementioned lack of adult-literacy training, suggests this problem will not go away soon.

Overall, learning disabilities affect an individual's ability to develop and use reading, writing, reasoning, and mathematical skills (Karande et al. 2005). Sherwani et al. (2009) reported that the effects of low literacy go beyond the processing of written text. Supporting this claim, low-literacy subjects in their study also experienced difficulty with tasks supported by speech interfaces. In addition, Viswanathan, Rosa, and Harris (2005) reported low-literacy consumers are more likely to employ concrete reasoning, basing decisions, trade-offs, and behaviors on the literal meaning of single pieces of information rather than a comprehensive consideration of all available information. Consistent with the use of concrete reasoning, they also report this population struggles with abstract or metaphorical meaning. In the absence of effective information-processing skills, the individual's ability to access, process, and retain (learn) information is severely constrained.

One point where the learning-disabled population breaks with the larger functionally illiterate community is at the level of underlying intelligence. Many individuals with one or more learning disabilities have normal or above normal intelligence (Rowland 2004; Doak, Doak, and Root 1996; Gerber 1998). Consequently, once appropriate accommodations are provided to mitigate the effects of the disability, adequate intelligence exists to accomplish most cognitive tasks.

A final, yet critical, defining characteristic is the incredible diversity within the population—often defined as a "universe of one." Individuals will vary in the range and severity of deficits and the range and type of compensating abilities. Ozannes, Adkins, and Sandlin (2005) suggest literacy skills are also highly contextual—not transferring smoothly from one situation to another. Although their research focused on low-literacy consumers, their findings warrant consideration by those conducting similar studies in HCI.

39.2.1 Low Literacy, Learning Disability, and Aging

One remaining characteristic of this population is the disproportionately high rates of lower literacy in the aging population. Kirsch et al. (1993b, p. 5) reported: "Older adults were more likely than middle-aged and younger adults to demonstrate limited literacy skills. For example, adults over the age of 65 have average literacy scores that range from 56 to 61 points (or more than one level) below those of adults 40–54 years of age." (The Literacy Skills of America's Adults). In the most recent National Literacy Survey, 14% of the adult population is below basic prose literacy skills, with 26% of this group 65 years of age or older (NAAL 2003). As cited in Section 39.1.3.1, the combination of low literacy and aging is particularly problematic when it comes to accurately following instructions, carefully taking medications, and properly using health technologies.

The cause for the disproportionately high percentage of low literacy among older adults is likely a combination of a number of factors including a lower percentage of high school graduates among the older population and the general lack of special education services when they were in school. Also, Patterson (2008) highlights the likelihood that the abilities of the learning disabled decline with age, exacerbated by the general cognitive decline associated with aging. Regardless of the cause, design accommodations that benefit lower-literacy populations are likely to have the added benefit of supporting the broader aging population as well.

The Section 39.2.2 will move from this general overview of population characteristics to a detailed discussion of the functional characteristics of the learning disabled. A sizable portion of the research that fuels this discussion is from the study of dyslexia. The rationale for this focus is twofold: (1) dyslexia is the most common of the learning disabilities, affecting nearly 80% of the learning-disabled population (Karande et al. 2005);

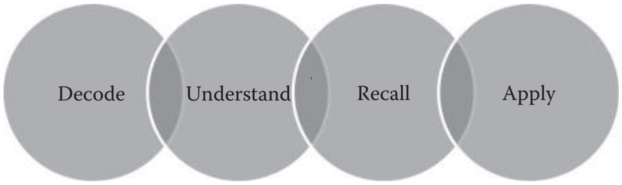


FIGURE 39.2 Four stages of information processing.

and (2) deficits associated with dyslexia are most likely to impact information processing as it relates to HCI.

39.2.2 Functional Characteristics of the Learning Disabled

Because of the previously cited diversity in the learning-disability community and the high degree of overlap among clinically diagnosed subgroups, it is easy to become overwhelmed by the complexity of the area. A more manageable strategy is to identify the functional characteristics shared across disabilities, with close attention to those that affect a particular interaction environment (Brown and Lawton 2001; Bohman and Anderson 2005). Figure 39.2 summarizes a number of proposed groupings.

After carefully examining these characteristics, four dynamically interconnected categories emerge as follows:

1. Reading
2. Memory
3. Metacognition
4. Search and navigation

These categories were selected because of their severity and their widespread effect on the HCI experience. In addition, many of the characteristics in Figure 39.2 are directly linked to deficits in a common underlying process, such as metacognition or working memory. These four categories, as highlighted in the discussion that follows, are not mutually exclusive. For example, skilled reading draws on memory and requires the deployment of metacognitive strategies. Similarly, search and navigation place a load on working memory, draw on long-term memory, and require metacognitive monitoring. The following section will review each of these categories more closely, and Section 39.3 will offer recommendations for accommodating each of these categories in system design.

39.2.2.1 Reading

Given the prominent role of written communication in everyday communication and the more demanding ICT environment, reading is one of the first barriers encountered by the low-literacy user. The alarming rate of poor prose and document-literacy skill was noted earlier. Also noted was the fact that poor reading is often tied to an underlying learning disability, inadequate schooling, or lack of exposure to reading. Typical in this area, an individual can suffer from all three conditions, two, or just one. As shown in Figure 39.2, reading is a multistaged process: word decoding, text processing, comprehension, recall, and application.

A reader could accurately process individual words yet fail to string these words together to achieve a larger meaning (comprehension). Readers can also demonstrate immediate comprehension of a passage yet fail to correctly apply that understanding to similar or different situations in the future. For the poor reader, the reading process is prone to breaking down at one or more of these stages. Mellard, Fall, and Woods (2010, p. 157) describe language comprehension as: “a complex construct that includes knowledge of vocabulary and information, as well as such higher-order abilities as recalling and sequencing events and making predictions and inferences.” The adult learners in their study, for example, failed to shift from a reliance on word recognition to language comprehension. In the following sections, connections are drawn between this complex act and deficits in metacognition and working memory among the learning disabled.

In dyslexic children, the disability affects reading at the most basic levels of phonological processing (Snowling, Deftry, and Goulandris 1996). Deficiencies at this basic level lead to further difficulties acquiring the complex skill of reading, building a rich vocabulary, and decreasing word-retrieval times. Most significantly, disabled readers allocate a disproportionately high amount of their attention to decoding letters and words, a process that is quickly automated by the nondisabled reader (van Gelderen et al. 2004). For the skilled reader, the process of automation frees the attention resource to focus on comprehension, the most critical component of the reading act. Skilled reading requires the management of a complex series of parallel actions, such as recognizing words, connecting those words to what came before, using those words to anticipate what will come next, and relating this combined experience to what one already knows. For the learning disabled, focusing most of the available attention on recognizing letters and decoding words severely compromises comprehension. Birru et al. (2004) reported that even when poor readers were able to read a passage, their inability to express answers related to the passage in their own words suggested minimal comprehension. Naturally, because reading is such a frustrating experience for the learning disabled, they are also less likely to read on a frequent basis, lowering the likelihood they will increase their reading skill through practice (Stanovich and West 1989).

Given the difficulty experienced by this population at the most basic level of decoding, it becomes clear that the problem is greatly exacerbated by materials written at a grade level beyond their ability. When information is presented at a level beyond the capability of the reader, mental workload is increased significantly and comprehension is greatly diminished. From an accessibility perspective, grade-level readability is a persistent problem throughout the web development community. In *E-Government for All*, Darrell West reports that 68% of state web sites and 70% of city sites are legible at the 12th-grade level (in Carvin, Hill, and Smothers [2004]). In an earlier study, West (2003, p. 3) found that

> 89% of government websites are not easily accessible to the citizenry because the site read at higher than an eighth grade

> level of literacy. Fully two-thirds of all sites have language consistent with a 12th-grade reading level, which is higher than the average American [half of all Americans read no higher than the eighth grade level]."

In another review conducted by Croft and Peterson (2002), the investigators found the mean readability score of 145 asthma-related websites was above the 10th grade, with 27 of the websites at the maximum 12th-grade level. Doak et al. (1998), summarizing a variety of studies, reported the average readability level of cancer information is near the 10th-grade level, and consent documents are written at the college/scientific level. Finally, Graber, D'Alessandro, and Johnson-West (2002) conducted a study of the readability level of privacy policies on Internet health websites. They found that the average readability of the policy was at a level equal to that of a second-year college student, far outside the reach of the average American. One large-scale study of the efficacy of a diabetes multimedia application produced disappointing results attributed in part to the readability of the supporting information. The investigators concluded that mere access to the technology was insufficient in the absence of appropriate design accommodations (Gerber et al. 2005).

Beyond basic readability criteria, Doak et al. (1998) documented the challenge presented by concept words (e.g., normal range, incidence), category words (e.g., angiotensin-converting enzyme inhibitors, chemotherapeutic agents), and value judgment terms (e.g., excessive, regularly). Finally, they reported mismatches in logic may also negatively affect compliance. For example, patients may feel it is appropriate to stop taking a medication once they start to feel well.

As previously noted, despite the presence of a reading disability, many in this population possess average or above average intelligence. As a consequence, if materials are presented at the appropriate readability level, the learning-disabled reader is capable of understanding. In separate studies conducted by Nielsen (2005) and Birru et al. (2004), poor readers performed well when interacting with easier-to-read materials.

Although there remains considerable disagreement on how best to measure grade-level reading, or even the appropriateness of such measures (Lewis 2010), certain measures and tests have gained wide acceptance and serve as a useful benchmark. One such test, the Flesch–Kincaid Grade-level readability test, measures readability by dividing the sentence length (number words divided by the number of sentences) by the average number of syllables per word (number of syllables divided by the number of words). The resulting calculation is then correlated with an appropriate grade level. Although this is a far from perfect measure, it does provide a useful benchmark of readability when evaluating the suitability of text for a given population. The Suitability Assessment of Materials instrument examines 22 factors related to reading difficulty. Unfortunately, this test along with others, such as the SMOG Readability Formula and the Fry Graph Reading Index, apply only to prose and not to lists, labels, and headings. With this type of text, the use of familiar words improves accessibility. Designers can consult reference works, such as the Word Frequency Book, (Carroll 1972) to identify the most commonly used words in the English language. Finally, Williams and Reiter (2008) examined text readability at the linguistic level focusing on the efficacy of cue phases, discourse ordering, and sentence structure. Clearly, readability is a complex and multilayered variable.

Because the functionally illiterate commonly experience difficulty with reading; illustrations, audio output, and video are often offered as effective accommodations. Carney and Levin (2002) summarize the general benefits of pictures, including increasing motivation, focusing attention, increasing depth of processing, clarifying text content, exploiting dual processing, and decreasing interference or decay. Houts et.al (2006) concluded that the inclusion of pictures in written and spoken health instructions increased low-literacy patients' attention, comprehension, recall, and adherence. Earlier, Houts et al (2001) found that the use of pictographs enhanced the recall of medical information over significant time periods. Although illustrations can assist the less-capable reader (Doak, Doak, and Root 1996; Weiner et al. 2004), positive effects vary with the reader and nature of the information. According to Beveridge and Griffiths (1987), illustrations contributed positively to reading performance in easier-to-read passages. In contrast, they also noted that illustrations degraded reading performance in the most difficult passages. Viswanathan, Rosa, and Harris (2005) found an over reliance on pictures among lower-literacy consumers, suggesting great care must be taken in the design and use of pictographic information.

In a study related to the supplemental use of animation, Larsen (1995) reported learning-disabled children had trouble due to the distraction of the animation. Similarly, Jiwnani (2001) reported system designers must be careful using auditory output since it has the potential to confuse the learning disabled. Jiwnani recommended the pace of the output must be slow, free of background noise, and repeatable under the control of the user. Finally, Hahn et al. (2004) demonstrated the positive outcomes of audio output for low-literacy users through their "talking touch screen" in a health care setting.

Clearly, interface and interaction designers must consider readability, illustrations, audio, and animation and their potential benefits for low-literacy users. Readability is the first barrier in design encountered by the low-literacy users. If this population is unable to process the language used in the display, all other accommodations are meaningless.

39.2.2.2 Memory

39.2.2.2.1 Long-Term Memory

The long-term and working-memory systems each have implications for the low literacy and learning-disabled population. As noted in Section 39.2.2.1, this population reads less often and fails to benefit from a major means of acquiring new knowledge. Without well-established domain models, interacting with web-based information or a software system is considerably more demanding. Doak et al. (1998) report that cancer information is most often organized according to the "Medical Model" doctors learned in medical school rather than models more appropriate for patients. This disconnect between doctor and patient models creates an often times insurmountable barrier to understanding and using information. Cognitive science has long recognized that long-term memories supplement and extend the limited capacity of working memory. This topic will be explored further in Section 39.2.2.4. Equally problematic is the tendency of the learning disabled to use the knowledge they do have without carefully evaluating its appropriateness, a metacognitive skill also discussed in Section 39.2.2.3 (Wilder and Williams 2001). In each of these cases, we see further evidence of the interconnected nature of these factors.

Although the underlying learning disability can make the acquisition of expertise more difficult, it would be wrong to think this population does not bring some level of learning and a variety of conceptual models to an interaction experience. As with so many other variables in accessibility design, we must recognize that the goals and models of the lower-literacy population may not align with those of the general population. Dickinson, Eisma, and Gregor (2003, p. 63) reported that: "users made remarks which indicated that they were not trying to understand the system or find generic rules that could help them to use it better." In other words, they were not attempting to understand the model underlying the system; they were simply trying to get something accomplished. System design should focus on that goal and avoid excessive functionality and extraneous information. From a development perspective, it is always easy to imagine occasions when one user or another might need this feature or that piece of information. Unfortunately, the lower-literacy population comes to the system with a greatly limited need, for both functions and information. Accessible design must consider these needs. Interface clutter increases mental workload on users as they attempt to locate information and features that align with their needs, a difficult challenge given their attentional impairments.

39.2.2.2.2 Working Memory

Cognitive science has long recognized the limited capacity of working memory and the major bottleneck it represents in information processing. Although learning disabilities vary in their clinical definition and in how they manifest themselves, one consistent variable in all learning disabilities is working-memory capacity. The dynamic interaction between memory capacity and the time information can be actively maintained imposes severe constraints on the learning disabled. Debate has raged for years whether the learning disabled suffer from diminished working-memory capacity or whether the complex dynamics of the underlying cognitive deficiencies place an excessive burden on a "normal" capacity (Daneman and Carpenter 1980; Poissant 1994; Swanson 1993; Vellutino et al. 1996; Ransby and Swanson 2003). Sabatini (2002) suggested that a combination of processing speed and working memory may affect the reading ability of adults with low literacy. Palmer (2000) suggests that dyslexic learners use phonological codes less efficiently, negatively affecting working-memory capacity. And finally, recognizing differences across the reading disability population, research by Swanson, Howard, and Saez (2006) found that different components of working memory explain the differences across various subgroups of reading disabilities. Sutcliffe, Fickas, and Sohlberg (2003) concluded that working-memory problems caused many users in their study to become disoriented when completing assigned tasks.

At the most basic level, the attention devoted to low-level decoding places a heavy demand on working-memory capacity. In addition, the increased likelihood of weaker mental models minimizes the opportunity to "offload" processing burden from the working memory, draw inferences, and make predictions. Further, the load imposed by metacognition, as discussed in Section 39.2.2.3, places additional demands on this limited resource. Finally, the high level of anxiety and frustration experienced by this population also operates in working memory and will negatively affect performance (Lee 1999). Anxiety is triggered by many variables including embarrassment, new situations, and fear of being labeled incompetent and testing situations.

In short, limited working-memory capacity accounts for the problems the learning disabled experience with search, decision making, sequenced operations in software, or in retracing their paths of travel navigating a website. When one considers the combined effects, it becomes clear why working memory, more than any other factor, guides for many of the design and support recommendations in Section 39.2.3. Interface and interaction designs must compensate for the limitations of working memory or suffer the consequences of lower performance, increased errors, and avoidance or abandonment of the task.

39.2.2.3 Metacognitive

Another critical deficiency in most dyslexics is in metacognition (Wilder and Williams 2001); the process of thinking about thinking (Flavell 1979). Metacognition takes many forms, including strategic planning, monitoring, self-appraisal, document-processing strategies, and reading strategies. Metacognition operates actively in working memory, parallel to the main information-processing task. Consequently, this activity is yet another process competing for limited resources. Also, because these strategies are learned behaviors, it suggests an interaction with long-term memory. Although nondisabled learners acquire most of these strategies without formal instruction, studies have shown these strategies can effectively be taught to the learning disabled (Collins et al. 1998; Wilder and Williams 2001). Unfortunately, for the small number of adults engaged in literacy training, most of the instruction is focused on decoding rather than higher-order comprehension and monitoring (Wilder and Williams 2001). Again, we must be reminded that skilled readers become skilled through years of practice with different reading materials that were processed to support diverse purposes. Because proficient readers and learners use these strategies on a frequent basis, they migrate, over time, to the level of an automatic skill. In contrast, the absence of persistent exposure to reading places the learning disabled at a further disadvantage.

Mokhtari and Reichard (2002) highlight the general agreement among researchers that awareness and monitoring of one's comprehension processes are a prerequisite for skilled reading. Cromley (2005) indicated that the learning disabled are less likely to know when they do not understand and are less likely to reread, synthesize, generate questions, or make predictions. They are also more likely to "satisfice" and accept partial or incorrect information rather than persevering to gain a more complete understanding. A distinguishing attribute of good readers is their ability to monitor the information-processing activity; in other words, they ask themselves: "Do I understand what I am reading? Do I see the relevance of the information to the task? What do I expect to follow in the next passage? How does this relate to what I already know?" Corley and Taymans (2002) organized this activity in three parts: (1) setting goals before the task and establishing a plan; (2) monitoring comprehension and understanding while engaging the task; and (3) evaluating one's learning after the task and making adaptations when faced with similar tasks. Danielson (2002), referencing users navigating the web, suggested that once goals are made, strategies are selected for achieving that goal. Further, strategies are constantly assessed and adjusted many times in a successful interaction experience. Given the mental demand of these activities, monitoring and adjusting activities pose a challenge for the disabled. Sutcliffe, Fickas, and Sohlberg (2003), working with cognitively disabled users, reported most of the identified usability problems could clearly be classified as metacognitive in nature. Finally, Britt and Gabrys (2002) indicated that document literacy requires four metacognitive skills: (1) sourcing (evaluating the credibility of source), (2) corroboration (seeking independent confirmation), (3) integration (creating mental representation of information), and (4) search (a skill woven throughout the other three areas).

The inability of the learning disabled to engage in self-evaluating activities is evident in the work of Birru et al. (2004). Although most of the participants in their study failed to answer the questions correctly, seven out of eight reported feeling "very comfortable" or "comfortable" with their Internet search experience. On the sourcing and corroboration skills, five out of eight used information provided on sponsored sites, yet still reported it was "very easy" to find trustworthy information on the Internet. Kruger and Dunning (1999) concluded that "less-skilled populations" lack the metacognitive ability to self-assess performance resulting in an overly inflated self-assessment. This finding has possible implications for any research conducted with this population that requires self-assessment of individual performance. Similarly, Viswanathan, Rosa, and Harris (2005) report that low-literacy populations display post-decision coping strategies whereby that rationalize outcomes to shift responsibility away from the self.

In short, metacognition requires a tremendous amount of cognitive activity, activity that occurs in parallel with the primary decoding task. All of this activity occurs in the executive control component of working memory. As noted in Section 39.2.2.2.2, this is a limited resource for the learning-disabled reader who devotes much of this resource to low-level decoding.

We take for granted that nondisabled users effectively and effortlessly engage in all of these activities. As noted in Section 39.3.4, design support for this variable amounts to providing constant feedback to the disabled user, explicitly connecting the information to the task, establishing checkpoints where the system queries the reader to monitor understanding, and engineering performance support in the form of cognitive scaffolding. On the topic of feedback, Dickinson, Eisma, and Gregor (2003) noted that longer tasks are tolerated by the learning disabled so long as they receive frequent feedback on what is happening and confirmation they are proceeding correctly. In the absence of such feedback, the poor reader becomes anxious, a state that further degrades performance since anxiety occupies the working-memory space. Finally, Sevilla et al. (2007) compared existing web content with its cognitively accessible—redesigned—equivalent. [In their population of cognitively disabled users], they found: "10 participants showed short-term memory problems when interacting with the conventional version and only 4 demonstrated such problems with the adapted version. This indicates how some cognitive deficits appear only when higher demands are made … another question can be asked with regard to the origins of the subject's navigational difficulties: specifically, is it a matter learning to navigate the web, or is it a matter of performing the navigation." (p. 20)

39.2.2.4 Navigation and Search

The learning disabled have traditionally experienced varying degrees of difficulty reading paper-based materials. They experience this difficulty despite the fact the linear format found in most paper-based materials offers considerable support for the reader. In these materials, the author assumes the burden for communicating organization through meaningful structures, logical sequences, helpful transitions, and explicit connections. In contrast, the nonlinear web requires the reader to infer organization and build a coherent model of the subject matter as the information unfolds in less predictable ways (Britt and Gabrys 2002). McDonald and Stevenson (1998) identified disorientation in the nonlinear environment as one of the most challenging elements of the web. Although disorientation is an obstacle that can be overcome by able users, this barrier is often insurmountable for the learning disabled. The source of this barrier is likely a combination of the working memory and metacognitive limitations previously discussed. This population, at the very least, experiences difficulty building a mental model of the information, monitoring where they are in the experience, or retracing their path. Each of these activities draws on their overtaxed working memory.

As one examines the factors affecting readers' ability to navigate, a number of variables emerge as follows:

- Basic navigation skills related to the interface
- Topology formats: nonlinear, hierarchical, and mixed topologies
- Depth versus breadth of the structure
- Navigational aids

39.2.2.4.1 Interaction Skills

Although there is limited research examining the effects of learning disabilities on navigation, the work of Zarcadoolas et al. (2002) suggested, with proper interventions, learning-disabled users can quickly learn and retain basic web-interaction skills. These skills include the behavior of links and the act of scrolling. Participants in this study did require training and prompting to use graphic links. Overall, the positive reaction to the linking convention is encouraging since they also found the participants, when given a choice, preferred following links rather than using search. Worth noting, Birru et al. (2004) found users in their study, although comfortable with the linking convention, would seldom click on more than one or two links to answer questions. Finally, although basic scrolling behavior was understood, none of the participants in the Zarcadoolas et al. (2002) study scrolled to view additional information. Without additional corroborating evidence, it is difficult to generalize from these limited studies. However, the behaviors exhibited in these studies warrant closer scrutiny as we continue to study and observe this class of user. Although not directly tied to low literacy, Sanchez and Wiley (2009) found that a scrolling format reduced comprehension of complex passages for readers with low working-memory capacity. As highlighted earlier, a diminished working-memory capacity is a common characteristic of the learning disabled. Given the possible connection between the action of scrolling and comprehension, the issue of scrolling also warrants further study.

In contrast to the relative ease with which this population learned basic interaction behaviors, a more significant challenge is faced while navigating the larger system. Danielson (2002) suggested users begin the navigation task with a decision about whether they are looking for a specific item, a group of items, or general information about the contents of a domain area. As noted in the metacognitive area, the learning disabled are generally weak at planning tasks, which immediately places them at a disadvantage in the navigation area. Unfortunately, there is little or no research that directly examines the learning-disabled population and the structural issues of nonlinear versus hierarchical, or depth versus breadth. In one study, Summers and Summers (n.d.) found low-literacy subjects, based on their desire to minimize reading, were better at linear text structures and deficient at nonlinear "way finding" tasks involving complex navigational structures. There is appropriate speculation on this issue since performance differences for general-population users are typically discussed within the context of working-memory capacity.

39.2.2.4.2 Topology

The work of McDonald and Stevenson (1998) shed some light on the issue of the efficacy of hierarchical, nonlinear, and mixed (hierarchical with referential links) topologies. Most significant perhaps for the purposes of this discussion, they found novices benefited most from the mixed structure since it offered a balance of freedom and control. The referential links supported exploration of a site without the support of a well-formed mental model, while the hierarchical framework served to constrain movements and minimize disorientation. Although the strict hierarchical structure provided the greatest control and guidance for participants, it proved inefficient when participants wanted to make distal movements in the structure. Finally, the nonlinear structure simply provided too many options and placed the heaviest demands on expertise and working memory. In an earlier study, McDonald and Stevenson (1996) found users stopped reading too soon when faced with too many decisions related to "what" and "how much" to read. In each of these situations, it is easy to speculate that the problems faced by the participants in these studies would only be exacerbated for the learning disabled, given their underlying deficiencies in working memory and metacognition.

39.2.2.4.3 Depth versus Breadth

The efficacy of breadth versus depth in the hierarchy is also open to speculation. As with topology, the depth issue is framed by limitations of working memory and the users' ability to distribute the navigation workload to their own conceptual models of the domain. In this discussion, we see greater diversity of opinion. In the world of web design, best practice favors breadth over depth, resulting in a greater number of choices at the highest level. Lewis (2007) discusses the challenges deep structures pose for the cognitively disabled. A study by Larson and Czerwinski (1998) suggested that a medium depth produced the best search performance over either the broader or the deeper options. Benard (n.d.) suggested depth versus breadth is not the best framing mechanism. Instead, he suggests the shape of the hierarchy is the best predictor of search performance, finding a broad first level, a narrow middle level, and a broad base to be most efficient. Although not based on a controlled study, Kolatch (2000) proposed a simpler top-level interface, offering fewer choices and a deeper structure for cognitively disabled users. Kolatch was careful to point out that this contradicts most research on the topic. Until research suggests otherwise, the balanced structure proposed by Larson and Czerwinski (1998) provides guidance consistent with our understanding of the adverse effects of too many choices and the mental load imposed by deep structures.

39.2.2.4.4 Navigation Aids

Danielson (2002) suggested that a variety of visual navigational aids can mitigate the aforementioned problems encountered in navigation. Navigational aids fall into two categories: (1) index and table of contents lists and (2) site maps. Danielson noted that graphical site maps are superior to lists and tables of contents at conveying the relationships between distal nodes on the site. The study also showed that the site maps benefited unknowledgeable users more than the knowledgeable. Finally, compared with a control group without site maps, Danielson reported that subjects with maps were less likely to abandon the task, reported information-seeking confidence, moved deeper into the site, and made more movements outside the hierarchy. Given the previously defined tendencies of the learning-disabled population, it is highly probable that they would benefit from the assistance offered by site maps. Supporting this theory, Mirchandani (2003) found detailed site maps helped the disabled user find information within a click or two.

39.2.2.4.5 Search

Research is slowly emerging related to the learning disabled and the search task. Not surprisingly, this research reports the learning disabled experience difficulty with search tasks. In an ethnographic study conducted by Zarcadoolas et al. (2002), the authors found a preference for following links versus using search. At the most basic level, the poor spelling displayed by many in the lower-literacy population creates a major barrier to successful searches. Search engines, such as Google, that offer alternative spellings are best. However, Sitbon and Bellot (2008) highlighted that dyslexic users are not assisted by the traditional spellchecker since they tend to write phonetically and sometimes group their words together. They suggested that this group requires special cognitive prosthetics designed to accommodate this tendency. Compounding search-related difficulties, weak self-monitoring skills also decrease the likelihood that the user will recognize that a spelling error was made. Even when search terms were spelled correctly, Birru et al. (2004) found that subjects rarely retyped search terms to locate more relevant items—a serious problem because they also found the low-literacy adults in this study rarely used optimal search terms as they attempted to retrieve health-related information. Birru et al (2004) concluded by highlighting the benefits of a categorizing search engine as one means of minimizing the adverse effects of poor search and long lists of results. This engine would sort results by category and minimize the need to evaluate long, unsorted lists. This is a good example of a technology assuming the burden for a weak metacognitive skill in the disabled population. Finally, Kodagoda and Wong (2008), in their inquiry of low-literacy search performance found the low-literacy subjects take eight times longer than high-literacy users on search tasks without producing a corresponding increase in search accuracy. Compared with their high-literacy counterparts, low-literacy users

- Spent one-third more time on each page visited
- Were far less focused as evident by the fact that they visited eight times more web pages
- Were 13 times more likely to backtrack
- Were four times more likely to revisit pages and 13 times more likely to be lost (p. 173)

39.2.3 Guidance for Research Methods and Testing

McGrenere, Sullivan, and Baeker (2006); Van Der Geest (2006); Wolf et al. (2006) each recommended including and accommodating the cognitively impaired in our research and evaluation methods. Since all of our methods involve communication, information processing, and interaction, it is reasonable to speculate that our methods require substantial modification consistent with the adjustments advocated throughout this chapter for design and support. As one reviews the functional characteristics highlighted in Section 39.2.2, it is likely that many of the necessary accommodations can be traced to metacognitive activities, workload demands on working memory, language difficulties, and the embarrassment associated with the disabilities.

The first challenge faced in working with this population is recruiting subjects. McGrenere, Sullivan, and Baeker (2006) suggest the variability within the cognitively disabled population makes it all but impossible to achieve a truly "representative" sample and challenges the notion of a typical or representative user. It also calls into question the notion of generalizability. Given this variability, it is disturbing that many of the studies purporting to support accessibility by including disabled users do so by mixing a small sample of cognitive and physical disabilities. Highlighting the challenge we face on the recruiting front, Ozanne, Adkins, and Sandlin (2005) distinguished between four classes of low-literacy consumers based on coping behaviors, with each behavioral class requiring different accommodations. At least one class, "the alienated consumer," exhibits highly developed avoidance and nondisclosure behaviors. One has to speculate whether this class would even participate in a study or if they did, would they reveal their disabilities? This is one of many variables we must consider.

On a practical level, recruiting low-literacy and cognitively disabled users requires considerably more time and effort. Utt (2010) suggested a 6–8-week lead time. Experience in our center has taught us that you experience a greater no-show rate with this population and you will experience a greater number of sessions that will need to be rejected for varied reasons. Consequently, always recruit extra participants to compensate for this potential loss. Our experience suggests that agencies and organizations that provide services to the adult-literacy community and cognitively disabled population are the most productive source for participants. Not only is this convenient, but many times they have done the formal testing and screening of clients. Establish a long-term relationship with these agencies so they will be assured you will treat their clients with dignity and respect. Viswanathan, Rosa, and Harris (2005) recommended longer term volunteer work with this population to build a relationship of trust and foster a heightened sensitivity to their needs. Finally, Brez and Taylor (1997) cautioned that care must be given to administering literacy screeners given the stigma attached to this disability. This was discussed in detail in Section 39.1.1. When forced to screen for literacy, the health care industry has developed a number of literacy screening tests such as the Wide Range Achievement Test, the Rapid Estimate of Adult Literacy in Medicine, and the Test of Functional Health Literacy in Adults. Finally, with more severely disabled populations, some researchers employ proxy users.

Unfortunately, caregivers do not always have perfect knowledge of the abilities, behaviors, and goals of users. Quite simply, their motivations are often different.

Great care must be taken to ensure appropriate informed consent. Recommendations on this issue vary, with the protocol within a particular group dictated by internal review boards (IRB). Generally speaking, consent forms should be written at the appropriate level and read aloud to the participant. Ask the participant to tell you, in their own words, what the form means before asking them to sign it. When in doubt about a participant's ability to provide informed consent, solicit the input of a counselor, friend, or caregiver.

There is an emerging pattern of findings suggesting that many of our traditional methods are inappropriate and ineffective for these populations. Small et al. (2005) reported their cognitively disabled participants exhibited difficulty with the think-aloud protocol, a strategy they quickly abandoned in favor of a guided walkthrough approach. Viswanathan, Rosa, and Harris (2005) cautioned against traditional research techniques such as interviews and surveys and recommended a more hermeneutical approach. Lepisto and Ovaska (2004) found the think-aloud protocol did not work well with the cognitively disabled population. Instead, they used a variety of techniques interviews, observations, expert reviews, and informal walkthroughs. Findings from each of these sources were compared and integrated in the analysis phase. Finally, the investigators also stressed the need to adjust each method to meet the highly individualized needs of the population.

A growing body of work with low literacy and cognitively disabled populations demonstrates the effectiveness of the ethnographic method (Zarcadoolas et al. 2002; Carmien and Fischer 2008). Dawe (2007) combined the ethnographic method with evolving technology probes (prototype). Dawe found the ability of subjects to interact with the probe helped to overcome subject's limited language skills. By observing users in natural settings, they could show and demonstrate rather than describe their needs and opinions through abstract or hypothetical scenarios. Dawe highlighted the critically important contribution of the technology probe by showcasing the users' tacit knowledge displayed through action. As noted in Section 39.2.2, this population exhibits difficultly with this class of thinking activity. Ethnographic study is emerging as one of the most productive research techniques based on its ability to mitigate the various communications and thinking barriers exhibited by this population.

Other investigators have also reported success with methods similar to ethnography including grounded theory, task artifact theory, and hermeneutics. Grounded theory is a systematic method for generating theoretical statements from case studies. Wolf et al. (2006) used a grounded theory approach to explore the basis for low-literacy patients' incorrect interpretations of design alternatives through an analysis of their verbatim responses. Based on cognitive interviews, grounded theory guided the inductive process of organizing content derived from patients' comments. For this study, patients' misinterpretations were reviewed and classified using both predetermined and emerging coding schemes. Sutcliffe, Fickas, and Sohlberg (2003) recommended a similar approach in the "task artifact theory," a framework that describes an iterative approach to research and design. "It asserts that well founded HCI designs should relate to theoretically grounded knowledge. The designs are produced after a detailed task analysis and application of existing theory; then the design is evaluated, revised, and design principles are extracted as psychologically motivated design rationale" (p. 578) Ozanne, Adkins, and Sandlin (2005) used an iterative hermeneutical approach in their study of low-literacy consumers, shifting back and forth between the data collected and the literature to identify a logical claim of evidence and arrive at a coherent framework. Finally, as noted above, Viswanathan, Rosa, and Harris (2005) recommended a more hermeneutical approach in their investigative work with this group of users. Although further consideration of these methods is warranted, the iterative nature of these methods, grounded in appropriate theory, likely compensates for the diversity and lack of predictability within this population.

Because of varied communication barriers, compounded by possible embarrassment and anxiety, traditional structured interview techniques are particularly challenging for low-literacy populations. Huenerfauth, Feng, and Elhadad (2009) found adults with intellectual disabilities had difficulty with simple yes/no questions and Likert-scale questions. In contrast, they noted positive outcomes with multiple choice questions with three answer choices, each illustrated with a simple illustration or photo. As noted elsewhere, this population has difficulty choosing among alternative and speculating on hypothetical or abstract situations. Milne, Clare, and Bull (1999), focusing on police interviewing techniques, discussed the relative merits of the cognitive interview technique with adults with mild learning disabilities. When compared with the structured interview technique, the cognitive interview enhanced recall of events but also produced an increase in confabulations. They reported that people with learning disabilities were significantly more likely than the general population to quickly submit to closed yes/no questions. Short-answer questions are also prone to incorrect answers with this population. The effectiveness of the cognitive interview is attributed to the use of cognitive mnemonics (Bekerian and Dennett 1993), which mitigate the previously described shortcomings. The technique uses four mnemonic strategies: (1) encourage the interviewee to restate the context, (2) report everything, (3) recall events in a different order, and (4) change perspectives.

There is strong support for integrating multiple methods as a means of overcoming the many challenges we face working with low-literacy users. One of these methods is the usability inspection. As documented earlier, many of the guidelines and inspection heuristics used in accessibility inspections are general and vague. Romen and Svanaes (2008), through controlled usability tests, reported that only 27% of the identified website accessibility problems could be identified through the WCAG produced through the W3C. They reviewed other studies producing similar results. This study confirms the inadequacy of current heuristics for expert reviews focused on accessibility and highlights the need for further work in this area.

Finally, there appears to be a number of issues surrounding our use of the think-aloud protocol and usability testing in general. As discussed previously, Small et al. (2005) reported their cognitively disabled participants exhibited difficulty with the think-aloud protocol. This problem most likely results from metacognitive deficiencies and the work load demands of the think-aloud protocol on limited working-memory resources. Traditional usability "testing" is also likely to trigger anxiety and further drain the working-memory resource. There also appears to be problems with the accuracy of self-reporting with this population, Sherwani et al. (2009) noted concerns for the inability of low-literacy subjects to accurately distinguish between information acquired through the system under examination and prior knowledge. They recommended that subjective feedback requires validation from other sources since, in their experience; this information is often unreliable for a multitude of reasons. For example, low-literacy users in their study reported a preference for the touchtone interface even though they performed better with the speech interface. There also seems to be concerns for the accuracy of performance and attribution as evident in the work of Birru et al. (2004) and Viswanathan, Rosa, and Harris (2005).

When testing is unavoidable, consider the appropriateness of timed tasks for a population with processing speed problems. Lepisto and Ovaska (2004) confirmed this concern, reporting that timed tasks do not work well with this population who often complete tasks at their own rate.

Tremendous progress has been made over the past 10 years understanding the shortcomings of our traditional methods. These deficiencies are not surprising since these methods were developed and refined over many decades working with the general population. In their place, we find promising new methods and suitable refinements of existing techniques that will help us better understand the requirements of this group and evaluate the appropriateness of our design solutions.

39.3 DESIGN ACCOMMODATIONS

Given the documented characteristics of this population, it is clear that design accommodations are necessary to make products accessible. Before moving into specific recommendations, based on the four-part classification strategy proposed earlier, we will first examine broader strategies for managing these accommodations. There are four generally accepted approaches to accommodating the needs of the low-literacy and learning-disabled population. Each of these recommendations is informed by the previously discussed characteristics of this population, as follows:

1. Assistive technologies: Building technologies customized to compensate for the disability
2. Layered design: Designating special sections of the website or ICT for a disabled population
3. Personalization: Preference profiles created once and then applied to different web resources (http://www.fluidproject.org/)
4. Universal design: Adopting design practices that enable the learning-disabled population while benefiting the larger population as well

The following sections will briefly review the first two categories and provide more detailed recommendations for the third category, universal design.

39.3.1 Assistive Technologies

A number of studies have demonstrated the benefits of assistive technologies for the cognitively disabled (Brown 1992; Cole and Dehdashti 1998; Newell and Gregor 2000; McGrenere et al. 2003). A typical application in this genre is a standalone computer-based educational system, designed to assist severely disabled children. Other, less intrusive accommodations, including software agents and embedded scaffolding, have been explored by Shaw, Johnson, and Ganeshan (1999); Quintana, Krajcit, and Soloway (2002); and Shneiderman (2003). Cole and Dehdashti (1998) and Carmien and Fischer (2008) refer to this class of technologies as cognitive prosthetics.

Scaffolding and agents offer performance support to the disabled user in the early stages of interaction with a new product. Shneiderman (2000) described this approach as "evolutionary learning." As the user becomes more proficient with the requirements of the system and task, the scaffolding is slowly torn down, either under the control of the user or through intelligent software agents. Most typically, the agent replaces or supports a deficient metacognitive process by assisting with planning, monitoring, self-appraisal, strategy setting, and feedback. Quite simply, this burden is shifted from the user's working memory to the system. Shneiderman (2000), Quintana, Krajcik, and Soloway (2002), and Carmien and Fischer (2008) highlighted the efficacy of this approach for supporting less-knowledgeable and disabled populations. Quintana, Zang, and Krajcik (2005) demonstrated the value of scaffolding to support the metacognitive aspects of online inquiry by helping users plan, monitor, and regulate the activity. Each of these supported activities were previously identified as deficient in this population. Finally, we see the increased development of systems such a SkillSum that adapt content, presentation, and linguistic structure to accommodate the needs of low-literacy readers (Williams and Reiter 2008).

39.3.2 Personalization

Personalization of the interface, interaction, and information design is increasingly recognized as an effective means of addressing the highly individualized needs of this population. Naturally, this would only work for technologies under the control of an individual. Publically accessed technologies will continue to look to the principles of universal design as the best available solution. Sutcliffe, Fickas, Sohlberg, and Ehlhardt (2003) investigated the efficacy of four different interfaces for a new e-mail system for cognitively disabled users. Although all users preferred interfaces that did not restrict their freedom, no one design proved superior. The authors saw these findings supporting the need to support customization of interfaces for this population. Similar findings were reported by Myatt, Essa, and Rogers (2000). While Petrie, Weber, and Fisher (2005) proposed the use of content management and personalization profiles to adapt content, interaction, and navigation to meet the diverse needs of print-disabled users.

39.3.3 Layered Design

Layered design strategy often requires the design of two versions of the system: one for the general population and one for low-literacy users. Although it is generally preferred not to separate the disabled and general populations, layered design is simply an extension of a practice that has been used for years for separating domain experts from non-experts or native speakers from nonnative speakers. In this case, one area of the site would be written at the population average eighth-grade level, whereas another section of the site would be written at the fifth-grade level or lower. Lobach et al. (2004) described the benefits of a two-tier literacy system in their study. Sites such as this may include other assistance, such as agents, site maps, illustrations, and the like. A variation of this strategy is the levels-of-reading approach where progressively deeper levels of detail are offered to the reader. Poor readers—and others for that matter—can choose the level appropriate to their needs and abilities. A more complex form of language support can be found on the Center for Applied Special Technology (CAST) website (http://www.cast.org/) where the sponsors provide a language tool to support low-literacy users visiting the CAST site.

39.3.4 Universal Design

The first principle of universal design dictates "provide the same means of use for all users: identical if possible—equivalent when not." In contrast to the first two approaches, universal design accommodations are seamlessly embedded in the system interface and interaction design. This class of accommodations is less likely to affect development costs or create an obstacle for the nondisabled. Consequently, these accommodations are more likely to be embraced by the wider development and user communities. Further, it is widely accepted in the accessibility community that these accommodations improve the usability of the product for all users (Shneiderman 2000; Dickinson, Eisma, and Gregor 2003). The following recommendations are organized in the previously defined categories of functional characteristics, although support for a given recommendation is often found in multiple categories. Finally, recommendations designated with an * identify recommendations that require additional research and warrant close monitoring. All others enjoy wide support in the literature.

39.3.4.1 Reading

When designing the language component of any system, support the following:

- Maintain an appropriate reading level (use readability formulas to establish a benchmark; measure word difficulty and sentence length).
- Use active voice.
- Limit information to the key information users need.
- Use the Word Frequency Book to identify common words.
- Place information and instruction in context.
- Employ lists.
- Avoid or recognize the consequences of using concept, categorizing, or value judgment words.
- Chunk information.
- Present content in sequence.
- Repeat information from screen to screen (do not assume the user will carry over).
- Maintain consistency in language and procedures.
- Communicate directly and concretely.
- Emphasize actions users must complete.
- Confirm actions are completed.
- Highlight critical information, information structure, or new information.
- Use familiar terms, and avoid acronyms and jargon.
- Use visual and auditory prompts.
- Provide definitions of critical terms through direct linking to glossary.
- Use illustrations to complement text, communicate structure, and emphasize connections.
- Highlight (using circles, arrows, and the like) critical areas of an illustration and explicitly connect to the text.
- Avoid the gratuitous use of animations and other movement.*
- Use 12-point type.
- Use familiar typefaces (there is conflicting research regarding the efficacy of serif versus sans serif).*
- Avoid tight letter-spacing (makes low-level decoding more difficult).
- Pace auditory output slowly, and allow user control to repeat output.
- Avoid background noise with auditory output.
- Provide "looser" versus tighter line spacing.
- Support easy, user controlled, style changes.
- Make it "look easy" by lowering information density.
- Limit use of italics and uppercase (effects are much more severe than for the general population).
- Maintain higher contrast.*
- Avoid light text on a dark background.
- Use ragged right formats to preserve consistent word space.

39.3.4.2 Memory

In an effort to decrease mental workload, support the following:

- Maintain consistency in all aspects of the design.
- Minimize anxiety.
- Leverage existing knowledge, behaviors, and tasks.
- Avoid splitting attention between two tasks.
- Focus on the user goals.
- Limit information and features to what is really needed.
- Limit chunking complexity for audio output (particularly interactive voice response [IVR]).
- Focus on behaviors and tasks rather than facts.
- Partition tasks in reasonably sized groups.
- Minimize scrolling.
- Avoid the need to retain information over long tasks or across multiple screens.
- Support mental calculations, decisions, and comparisons.
- Complete mathematical calculations.
- Limit choices.
- Provide a list of options for entry fields.
- Complete information automatically in forms and fields whenever possible.
- Use advanced organizers.
- Use mnemonics.
- Minimize screen clutter.
- Provide extra time for tasks.
- Eliminate the anxiety of timeouts.

39.3.4.3 Metacognitive

In an effort to strengthen or replace deficient metacognitive processes, support the following:

- Maintain a consistent design.
- Provide guidance, status reports, and feedback.
- Use headings to identify critical information.
- Convey associations between new information or process and that which is known.
- Communicate goal or purpose of the site immediately.
- Communicate prerequisite knowledge for the task and provide convenient links.
- Communicate required sequences or organizational structures.
- Allow the user to interact with the information.
- Use checklists to support self-monitoring.
- Avoid choices that require fine discriminations or close monitoring.
- Design for immediate/early success (which lowers anxiety and builds confidence).
- Align with user goals, and provide reward or convey value proposition (motivation).
- Support cooperative work activities.
- Provide reminders.
- Communicate task status.
- Provide source information for material presented.
- Minimize embedded links.
- Avoid taking user to other pages for ancillary information.
- Use error prevention and recovery support.
- Query when choices or decisions are required.
- Review information entered or validate the successful completion of a process.
- Use auditory and visual cues to mark stages of a work cycle, helping the user self-monitor.

39.3.4.4 Navigation and Search

In an effort to improve the effectiveness of search and navigation, support the following:

- Make information and features supporting goals readily accessible to minimize navigation and search.
- Provide persistent presentation of path history.
- Limit distractions.
- Offer persistent opportunity to exit, backup, or return to start.
- Provide status indicators.
- Minimize scrolling.
- Label all links.
- Provide linked paths supporting probable scenarios.
- Maintain alerts onscreen until dismissed by user.
- Place information or process in context.
- Partition information into categories defined by clear rules.
- Use site maps, tables of contents, and indexes.
- Use a topology that is primarily hierarchical with referential links to areas aligned with goals.*
- Maintain a medium breadth and depth—not too broad, not too deep.*
- Use clear, thematic labels at each level.
- Provide productive terms for search.
- Offer suggested spellings.
- Use categorizing search engines.
- Provide performance support for evaluating search results (corroboration and sourcing).

39.3.4.5 Guidance for Research and Usability Testing

To produce the most reliable data and protect the well-being of participants, support the following:

- Consider ethnographic studies rather than structured interviews.
- Use cognitive interview technique.
- Use grounded theory or hermeneutical research methods.
- Consider technology probes to provide observational opportunities while reducing reflection demands.
- Avoid embarrassment or humiliation (anxiety lowers performance).

- Greatly increase recruiting time.
- Build relationships with organizations working with this population.
- Include councilors or caregivers with the more severely disabled.
- Consider ethical obligations when screening for lower-literacy samples (full-disclosure issues, informed consent, and IRB).
- Minimize anxiety in testing situation.
- Test in context (actual work environment).
- Avoid using the term "testing" since this population equates the term with failure.
- Break tasks in the test script into small, yet logical, units.
- Avoid questions requiring "yes" or "no" answers, such as "Do you understand?"
- Avoid Likert-scale questions.
- Consider the appropriateness of timed tasks.
- Provide multiple choice questions with visual support for each answer.
- Require users to repeat back their understanding of information in their own words.
- Shorten test times to minimize effects of fatigue on performance and to compensate for attention deficit.
- Use direct interaction protocol since users may not freely think aloud (inadequate working memory to support both think-aloud and to acclimate to the task).*
- Accommodate slower processing speeds when setting performance levels (e.g., task times).
- Expect participants to report exaggerated levels of performance or problems attributed beyond self as a means of covering poor literacy skills.
- Invite a trusted caregiver to any session when working with more severely cognitively disabled users.

39.4 CONCLUSION

Fifteen years ago, our concern for the effects of lower literacy was confined exclusively to the processing of print media or spoken instructions. With the explosion of information technology in the workplace and the role of the Internet as the repository for information of all types, literacy has become a major barrier for millions of citizens as they compete for work, attempt to improve the quality of their lives, and participate in civic activities. Although the challenges faced by low-literacy populations with print media are significant, the problems become greatly exacerbated with the increased mental workload imposed by ICT.

As advocated in this chapter, many of the barriers to accessibility require simple modifications in the interface and interaction design. Accessible design is an art, in which the needs of one group are carefully balanced against those of another, and the designer recognizes that "accessible for most" is more achievable than "accessible for all." As one reviews the list of design accommodations required by the low-literacy population, HCI professionals will recognize most, if not all, of these variables as ones considered in each and every interface design. The difference—and a significant one—is that fully capable users exhibit flexible learning, problem solving, and interaction skills that allow them to adapt to less-than-perfect designs. The learning disabled, in contrast, lack cognitive flexibility and suffer varying degrees of performance degradation in nonoptimum design environments. By increasing our understanding of this sizable population and embracing the design practices outlined here, we may finally ensure that universal accessibility is truly universal.

REFERENCES

Adkins, N., and J. Ozanne. 2005. The low literate consumer. *J Consum Res* 32:93–105.

Bekerian, D., and J. Dennett. 1993. The cognitive interview technique: Reviving the issues. *Appl Cogn Psychol* 7:275–97.

Benard, M. (n.d.). *Criteria for Optimal Web Design*. http://psychology.wichita.edu/optimalweb/structure.html (accessed January 13, 2006).

Benton Foundation. 1999. *Networking for Better Care: Health Care in the Information Age*. htpp://www.Benton.org/library/health/ (accessed December 28, 2003).

Beveridge, M., and V. Griffiths. 1987. The effects of pictures on the reading processes of less able readers: A miscue analysis approach. *J Res Read* 10:29–42.

Birru, M., V. Monaco, L. Charles, H. Drew, V. Njie, T. Bierria, E. Detlefsen, and R. Steinman. 2004. Internet usage by low-literacy adults seeking health information: An observational analysis. *J Med Internet Res* 6(3). http://www.jmir.org/2004/3/e25/ (accessed December 12, 2005).

Bogner, M. 1999. How do I work this thing? Cognitive issues in home medical equipment use and maintenance. In *Processing of Medical Information in Aging Patients: A Cognitive and Human Factors Perspective*. ed. R. Morrell, K. Shifren, and D. Park, 245–56. New York: Psychology Press.

Bohman, P., and S. Anderson. 2005. A conceptual framework for accessibilty tools to benefit users with cognitive disabilities. In *Proceedings of the W4A at WWW2005*, 85–9. New York: ACM.

Brez, S., and M. Taylor. 1997. Assessing literacy for patient teaching: Perspectives of adults with low literacy skills. *J Adv Nurs* 25:1040–7.

Britt, M., and G. Gabrys. 2002. Implications of document-level literacy skills for website design. *Behav Res Methods Instrum Comput* 34(2):170–6.

Brown, C. 1992. Assistive technology computers and persons with disabilities. *Commun ACM* 35(5):36–45.

Brown, D., and J. Lawton. 2001. Design guidelines and issues for website design and use by people with a learning disability. Centre for Educational Technology Interoperability Standards. http://www.cetis.ac.uk/members/accessibility/links/disabilities/cogdis (accessed March 28, 2007).

Cain, M. M., R. Mittman, J. Sarasohn-Kahn, and J. Wayne. 2000. *Health E-People: The Online Consumer Experience*. Oakland, CA: Institute for the Future, California Health Care Foundation.

Carmien, S., M. Dawe, G. Fischer, A. Gorman, A. Kintsch, and J. Sullivan. 2005. Socio-technical environments supporting people with cognitive disabilities using public transportation. *ACM Trans Hum Comput Interact* 12(2):233–67.

Carmien, S., and G. Fischer. 2008. Design, adoption, and assessment of a socio-technical environment supporting independence for persons with cognitive disabilities. In *Proceedings of CHI*, 597–606. New York: ACM.

Carney, R., and J. Levin. 2002. Pictorial illustrations still improve students' learning from text. *Educ Psychol Rev* 14(1):5–26.

Carroll, J. 1972. *The American Heritage Word Frequency Book*. New York: Houghton Mifflin.

Carvin, A., J. Hill, and S. Smothers. 2004. *E-Government for All: Ensuring Equitable Access to Online Government Services*. New York: The EDC Center for Media & Community and the NYS Forum.

Cashen, M., P. Dykes, and B. Gerber. 2004. e-health technology and Internet resources: Barriers for vulnerable populations. *J Cardiovasc Nurs* 19(3):209–14.

Cole, E., and P. Dehdashti. 1998. Computer-based cognitive prosthetics: Assistive technology for the treatment of cognitive disabilities. In *Proceedings of the Third International ACM Conference on Assistive Technologies*, 11–8. Arrington, VA: ACM.

Collins, V., S. Dickson, D. Simmons, and E. Kameenui. 1998. *Metacognition and its Relations to Reading Comprehension: A Synthesis of the Research. National Center to Improve the Tools of Educators*. Washington, DC: University of Washington.

Corley, M., and J. Taymans. 2002. *Adults with Learning Disabilities: A Review of the Literature*. http://www.ncsall.net/?id1575 (accessed December 30, 2005).

Craven, J., and H. Booth. 2006. Putting awareness into practice: Practical steps for conducting usability tests. *Libr Rev* 55(3):179–94.

Croft, D. R., and M. W. Peterson. 2002. An evaluation of the quality and contents of asthma education on the World Wide Web. *Chest* 121(4):1301–7.

Cromley, J. 2005. *Metacognition. Cognitive Strategy Instruction, and Reading in Adult Literacy*. http://www.ncsall.net/index.php?id5773 (accessed January 17, 2006).

Daneman, M., and P. Carpenter. 1980. Individual differences in working memory and reading. *J Verbal Learn Verbal Behav* 19:450–66.

Danielson, D. 2002. Web navigation and the behavioral effects of constantly visible site maps. *Interact Comput* 14(5):601–18.

Dawe, M. 2007. "Let me show you what i want": Engaging individuals with cognitive disabilities and their families in design. In *Proceedings of Computer Human Interaction Conference*, 2177–82. New York: ACM.

Dickinson, A., R. Eisma, and P. Gregor. 2003. Challenging Interfaces/Redesigning Users. In *CUU 2003*, 61–8. British Columbia, Canada. New York: ACM.

Doak, C., L. Doak, G. Friedell, and C. Meade. 1998. Improving comprehension for cancer patients with low literacy skills: strategies for clinicians. *CA Cancer J Clin* 48:151–62.

Doak, C., L. Doak, and J. Root. 1996. *Teaching Patients with Low Literacy Skills*. Philadelphia, PA: J.B. Lippincott Company.

Dropout Rates in the United States. 2001. National Center for Educational Statistics. http://nces.ed.gov/pubs2002/dropoutpub_2001/11 (accessed December 10, 2003).

Epstein, A., R. Maley, and J. Suri. 2006. A tutorial on emerging medical device technology in the distributed diagnosis and home healthcare (D2H2) W+Environment. In *Workshop, 28th Annual International Conference IEEE Engineering in Medicine and Biology Society (EMBS) August 30-September 3*. New York: IEEE. http://embc2006.nijit.edu/ws10php (accessed June 10, 2010).

Flavell, J. 1979. Metacognition and cognitive monitoring: A new area of cognitive-developmental inquiry. *Am Psychol* 34(10):906–11.

Friedman, M., and D. Bryen. 2007. Web accessibility design recommendations for people with cognitive disabilities. *Technol Disabil* 19:205–12.

Gerber, P. 1998. *Characteristics of Adults with Specific Learning Disabilities*. http://www.idonline.org/ld_indepth/adult/characteristics.html (accessed December 28, 2003).

Gerber, B., I. Brodsky, K. Lawless, L. Smolin, A. Arozullah, E. Smith, M. Berbaum, P. Heckerling, and A. Eiser. 2005. Implementation and evaluation of a low literacy diabetes education computer multimedia application. *Diabet Care* 28(7):1574–80.

Graber, M. A., D. M. D'Alessandro, and J. Johnson-West. 2002. Reading level of privacy policies on Internet websites. *J Fam Pract* 51(7):642–5.

Gribbons, W. 1992. The functionally illiterate: Handicapped by design. In *Proceedings of the International Professional Communications Conference*, 302–7. New York: IEEE.

Hahn, E., D. Cella, D. Dobrez, G. Shiomoto, S. Marcus, M. Vohra, H. Chang, and K. Webster. 2004. The talking touchscreen: A new approach to outcomes assessment in low literacy. *Psychooncology* 13:86–95.

Henriksen, K., A. Joseph, and T. Zayas-Caban. 2009. The human factors of home health care: A conceptual model for examining safety and quality concerns. *J Pat Saf* 5(4):229–36.

Houts, P., C. Doak, L. Doak, and M. Loscalzo. 2006. The role of pictures in improving health communication: A review of research on attention, comprehension, recall, and adherence. *Patient Educ Couns* 61:173–90.

Houts, P., J. Witmer, M. Egeth, M. Loscalzo, and J. Zabora. 2001. Using pictographs to enhance recall of spoken medical instructions. *Patient Educ Couns* 43:231–42.

Huenerfauth, M. 2002. Design approaches for developing user interfaces accessible to illiterate users. In *American Association of Artificial Intelligence Conference (AAAI 2002), Intelligent and Situation-Aware Media and Presentations Workshop*. Paper presented in Edmonton, Alberta, Canada. Palo Alto, CA: Association for the Advancement of Artificial Intelligence.

Huenerfauth, M., L. Feng, and N. Elhadad. 2009. Comparing evaluation techniques for text readibility software for adults with intellectual disabilities. In *Proceedings of ASSETS '09*, 3–10. New York: ACM.

Hussey, L. 1994. Minimizing the effects of low literacy on medication knowledge and compliance among the elderly. *Clin Nurs Res* 3(2):132–45.

Jiwnani, K. 2001. *Designing for Users with Learning Disabilities*. http://www.otal.umd.edu/uupractice/cognition/ (accessed December 28, 2005).

Kaphingst, K., C. Zanfini, and K. Emmons. 2006. Accessibility of web sites containing colorectal cancer information to adults with limited literacy. *Cancer Causes Control* 17:147–51.

Karande, S., S. Sawant, M. Kulkarni, P. Galvankar, and R. Sholapurwala. 2005. Comparison of cognitive abilities between groups of children with specific learning disability having average, bright normal and superior nonverbal intelligence. *Indian J Med Sci* 59(3):95–103.

Kickbusch, I. 2001. Health literacy: Addressing the health and education divide. *Health Promot Int* 16(3):289–97.

Kirsch, I., A. Jungeblut, L. Jenkins, and A. Kolstad. 1993a. *Adult Illiteracy in America: A First Look at the Results of the National Literacy Survey*. Washington, DC: Department of Education, Center for Educational Statistics.

Kirsch, I., A. Jungeblut, L. Jenkins, and A. Kolstad. 1993b. *Executive Summary of Adult Literacy in America: A First Look at the Results of the National Literacy Survey*. http://www.nces.ed.govnaal/resources/execsumm.asp (accessed October 27, 2003).

Kodagoda, N., and B. L. Wong. 2008. Effects of low & high literacy on user performance in information search and retrieval. In *Proceedings of the 22nd British HCI Group Annual Conference on HCI 2008: People and Computers XXII: Culture, Creativity, Interaction—Vol 1. British Computer*, 173–81. London, UK.

Kolatch, E. 2000. *Designing for Users with Cognitive Disabilities*. http://www.otal.umd.edu/uuguide/erica/ (December 28, 2005).

Kruger, J., and D. Dunning. 1999. Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *J Personal Soc Psychol* 77(6):1121–35.

Larsen, S. 1995. What is "quality" in the use of technology for children with learning disabilities? *Learn Disabil Quarter* 18(2):118–30.

Larson, K., and M. Czerwinski. 1998. Web page design: Implications of memory, structure and scent for information retrieval. In *Proceedings of CHI '98*, 25–32. USA: ACM.

Lee, J. 1999. Test anxiety and working memory. *J Exp Educ* 67: 218–40. Retrieved December 4, 2005, from ProQuest database.

Lepisto, A., and S. Ovaska. 2004. Usability evaluation involving participants with cognitive disabilities. In *Proceedings of Nordi CHI*, 416–20. New York: ACM.

Lewis, C. 2006. HCI and cognitive disabilities. *Interactions* 13(3): 14–5.

Lewis, C. 2007. Simplicity in cognitive assistive technology: A framework and agenda for research. *Univers Access Inf Soc* 5:351–61.

Lewis, C. 2010. ICT and people with cognitive disabilities. *Interactions* 9(2):4–6.

Lobach, D., J. Arbanas, D. Mishra, M. Campbell, and B. Wildemuth. 2004. *Adapting the Human-Computer Interface for Reading Literacy and Computer Skill to Facilitate Collection of Information Directly from Patients. MEDINFO 2004*, 1143–6. Amsterdam, The Netherland: IOS Press.

McDonald, S., and R. Stevenson. 1996. Disorientation in hypertext: The effects of three text structures on navigation performance. *Appl Ergon* 27:61–8.

McDonald, S., and R. Stevenson. 1998. Effects of text structure and prior knowledge of the learner on navigation in hypertext. *Hum Factors* 40(1):18–27.

McGrenere, J., D. Davies, L. Findlater, P. Graf, M. Klawe, K. Moffatt, B. Purves, and S. Yang. 2003. Insights from the aphasia project: Designing technology for and with people who have aphasia. In *CUU 2003*, 112–8. Arlington, VA: ACM.

McGrenere, J., J. Sullivan, and R. Baekcker. 2006. Designing technology for people with cognitive impairments. In *Proceedings of CHI*, 1635–8. New York: ACM.

Mellard, D., E. Fall, and K. Woods. 2010. A path analysis of reading comprehension for adults with low literacy. *J Learn Disabil* 43(2):154–65.

Milne, R., I. Clare, and R. Bull. 1999. Using cognitive the interview with adults with mild learning disabilities. *Psychol Crime Law* 5:81–9.

Mirchandani, N. 2003. *Web Accessibility for People with Cognitive Disabilities: Universal Design Principles at Work*. http://www.ncddr.org/du/researchexchange/v08n03/8_access.html (accessed December 28, 2005).

Mokhtari, K., and C. Reichard. 2002. Assessing students' metacognitive awareness of reading strategies. *J Educ Psychol* 94(2):249–59.

Moon, R., T. Cheng, K. Patel, and P. Scheidt. 1998. Parental literacy level and understanding of medical information. *Pediatrics* 102(2). Retrieved from Medline, e25.

Myatt, A., I. Essa, and W. Rogers. 2000. Increasing the opportunities for aging in place. In *Proceedings of the ACM Conference on Universal Usability*, 65–71. New York: ACM Press.

National Assessment of Adult Literacy (NAAL). 2003. *A First Look at the Literacy of Americas Adults in the 21st Century*. Washington, DC: National Center for Educational Statistics, U.S. Department of Education, Institute of Educational Sciences, NCES 2006-470.

National Dyslexia Association (n.d.). *What is Dyslexia?* http://www.interdys.org (accessed November 10, 2003).

Newell, A., A. Carmichael, P. Gregor, and N. Alm. 2003. Information technology for cognitive support. In *The Human Computer Interaction Handbook*, ed. J. Jacko and A. Sears, 464–81. Mahwah, NJ: Lawrence Erlbaum Associates.

Newell, A., and P. Gregor. 2000. User sensitive inclusive design: In search of a new paradigm. In *CUU 2000*, 39–44. Arlington, VA: ACM.

Nielsen, J. 2005. *Lower-Literacy Users*. http://www.useit.com/alertbox/20050314.html (accessed December 9, 2005). Freemont, CA: Nielsen Norman Group.

Ozanne, J., N. Adkins, and J. Sandlin. 2005. Shopping for power: How adult literacy learners negotiate the marketplace. *Adult Educ Quarter* 55(4):251–68.

Palmer, S. 2000. Phonological recoding deficit in working memory of dyslexic teenagers. *J Res Read* 23(1):28–40.

Patterson, M. 2008. Learning disability prevalence and adult education program characteristics. *Learn Disabil Res Pract* 23(1):50–9.

Petrie, H., G. Weber, and W. Fisher. 2005. Personalization, interaction, and navigation in rich multimedia documents for print-disabled users. *IBM Syst J* 44(3):629–35.

Poissant, H. 1994. Assessing and understand the cognitive and metacognitive perspectives of adults who are poor readers. Center for the Study of Reading. Urbana, IL. Technical Report No. 594.

Quintana, C., J. Krajcik, and E. Soloway. 2002. A case study to distill structural scaffolding guidelines for scaffolded software environments. In *CHI 2002*, 81–8. Minneapolis, MN: ACM.

Quintana, C., M. Zhang, and J. Krajcik. 2005. A framework for supporting metacognitive aspects of online inquiry through software-based scaffolding. *Educ Psychol* 40(4):235–44.

Ransby, M., and H. L. Swanson. 2003. Reading comprehension skills of young adults with childhood diagnoses of dyslexia. *J Learn Disabil* 36(6):538–55.

Romen, D., and D. Svanaes. 2008. Evaluating website accessibility: Validating the WAI guidelines through usability testing with disabled users. In *Proceedings of NordiCHI*, 535–8. New York: ACM.

Rowland, C. 2004. *Cognitive Disabilities Part 2: Conceptualizing Design Considerations*. http://www.webaim.org/techniques/articles/conceptualize/?templatetype53 (accessed December 28, 2005).

Rudd, R. E., B. A. Moeykens, and T. C. Colton. 1999. *Health and Literacy: A Review of Medical and Public Health Literature*. New York: Josey-Bass.

Sabatini, J. 2002. Efficiency in word reading of adults: Ability group comparisons. *Scientific Studies of Reading* 6:267–98.

Sanchez, C., and J. Wiley. 2009. To scroll or not to scroll: Scrolling, working memory capacity, and comprehending complex texts. *Hum Factors* 51:730–8.

Sevilla, J., G. Herrera, B. Martinez, and F. Alcantud. 2007. Web accessibility for individuals with cognitive deficits: A comparative study between an existing commercial web and its cognitively accessible equivalent. *ACM Trans Hum Comput Interact* 14(3):12–25.

Shaw, E., W. Johnson, and R. Ganeshan. 1999. Pedagogical agents on the web. In *Proceedings of Autonomous Agents*, 283–90. Seattle, WA. New York: ACM.

Shaywitz, S. 1998. Current concepts: Dyslexia. *New England J Med* 338:307–12.

Sherwani, J., S. Palijo, S. Mirza, T. Ahmed, N. Ali, and R. Rosenfeld. 2009. Speech vs. touchtone: Telephony interfaces for information access by low literate users. In *Proceedings IEEE/ACM International Conference on Information and Communication Technologies and Development*, 1–11. New York: IEEE.

Shneiderman, B. 2000. Universal usability. *Commun ACM* 43(5):84–91.

Shneiderman, B. 2003. Promoting universal usability with multilayer interface design. In *CUU 2003*, 1–8. British Columbia, Canada: Vancouver.

Shneiderman, B. 2007. Web science: A provocative invitation to computer science. *Commun ACM* 50(6):25–7.

Sitbon, L., and P. Bellot. 2008. How to cope with questions typed by dyslexic users. In *Proceedings of the Second Workshop on Analytics for Noisy Unstructured Text Data. (Singapore, July 24, 2008)*, 1–8. New York, NY: ACM.

Small, J., P. Schallau, K. Brown, and R. Appleyard. 2005. Web accessibility for people with cognitive disabilities. In *Proceedings of CHI*, 1793–6. New York: ACM.

Snowling, M., N. Defty, and N. Goulandris. 1996. A longitudinal study of reading development in dyslexic children. *J Educ Psychol* 88(4):653–69.

Stanovich, K., and R. West. 1989. Exposure to print and orthographic processing. *Read Res Quarter* 24:402–33.

Stewart, R., V. Narendra, and A. Schmetzke. 2005. Accessibility and usability of online library databases. *Library Hi Tech* 23(2):265–86.

Stone, R., and J. Wiener. 2001. *Who will Care For Us? Addressing the Long-Term Care Workforce Crisis*, 1–44. The Urban Institute and American Association of Homes and Services for the Aging. Washington, DC: The Urban Institute.

Sum, A. 1999. *Literacy in the Labor Force*. Washington, DC: National Center for Education Statistics, U.S. Department of Education.

Summers, K., and M. Summers. (n.d.). *Reading and Navigational Strategies of Web users with Low Literacy Skills*. http://iat.ubalt.edu/summers/papers/Summers_ASIST2005.pdf (accessed May 12, 2010).

Sutcliffe, A., S. Fickas, M. Sohlberg, and L. Ehlhardt. 2003. Investigating the usability of assistive user interfaces. *Interact Comput* 15:577–602.

Swanson, H. L. 1993. Working memory in learning disability subgroups. *J Exp Child Psychol* 56:87–114.

Swanson, H. L., C. Howard, and L. Saez. 2006. Do different components of working memory underlie different subgroups of reading disabilities? *J Learn Disabil* 39(3):252–69.

TRACE Center. 1994. *A Brief Introduction to Disabilities*. Madison, WI: University of Wisconsin. htpp://trace.wisc.edu/docs/population/populat.htm (accessed November 23, 2003).

U.S. Department of Commerce. 2002. *A Nation Online: How Americans are Expanding Their Use of the Internet. Economics and Statistics Administration, National Telecommunications and Information Administration*. Washington, DC: NTIA.

Utt, M. 2010. Usability testing by people with disabilities. *Interactions* 9(2):18–9.

Van Der Geest, T. 2006. Conducting usability studies with users who are elderly or have disabilities. *Tech Commun* 53(1):23–31.

van Gelderen, A., R. Schoonen, K. de Glopper, J. Hulstijn, A. Simis, P. Snellings, and M. Stevenson. 2004. Linguistic knowledge, processing speed, and metacognitive knowledge in first and second language reading comprehension: A component analysis. *J Educ Psychol* 96(1):19–30.

Vellutino, F., D. Scanlon, E. Sipay, S. Small, R. Chen, A. Pratt, and M. Denckla. 1996. Cognitive profiles of difficult to remediate and readily remediated poor readers: Early intervention as a vehicle for distinguishing between cognitive and experimental deficits as basic causes of specific reading disability. *J Educ Psychol* 88(4):601–38.

Viswanathan, M., J. Rosa, and J. Harris. 2005. Decision making and coping of functionally illiterate consumers and some implications for marketing management. *J Mark* 69:15–31.

W3C. 1999. *Web Content Accessibility Guidelines 1.0 [Electronic-Edition].* http://www.w3.org/TR/WAI-WebContent/ (accessed April 3, 2007).

Weiner, J., A. Aquirre, K. Ravenell, K. Kovath, L. McDevit, J. Murphy, D. Asch, and J. Shea. 2004. Designing an illustrated patient satisfaction instrument for low literacy populations. *Am J Manag Care* 10(11):853–9.

West, D. 2003. *State and Federal E-Government in the United States. Center for Public Policy*. 1–29. Providence, RI: Brown University.

Wilder, A., and J. Williams. 2001. Students with severe learning disabilities can learn higher order comprehension skill. *J Educ Psychol* 93(2):268–78.

Williams, S., and E. Reiter. 2008. Generating readable texts for readers with low basic skills. *Nat Lang Eng* 14(4):495–525.

Wolf, M., T. Davis, H. Tilson, P. Bass, and R. Parker. 2006. Misunderstanding of prescription drug warnings labels among patients with low literacy. *Am J Syst Pharm* 63:1048–55.

Zarcadoolas, C., M. Blanco, J. Boyer, and A. Pleasant. 2002. Unweaving the web: An exploratory study of low-literate adults' navigation skills on the World Wide Web. *J Health Commun* 17:309–24.

40 Computing Technologies for Deaf and Hard of Hearing Users

Vicki L. Hanson

CONTENTS

40.1 INTRODUCTION

Traditionally, human interaction with computers has relied most heavily on visual perception and motor ability. Graphical user interfaces (GUIs) have dominated computer displays, both large and small, while mouse and keyboard devices have dominated computer input. Increasingly, however, newer interfaces are using sound and speech. The use of attention-grabbing multimedia computer presentations plus the increasing use of conversational speech interfaces are examples.

As exciting as these new technologies may be, they have the potential to disenfranchise deaf and hard of hearing users. This chapter will discuss interface technologies as they relate to the needs of users unable to hear auditory information. In addition, language considerations associated with limited hearing will be discussed.

This chapter begins with a discussion of hearing loss, followed by issues of language acquisition as they relate to hearing loss. Auditory user interfaces that present difficulties will be discussed next, along with information about interface alternatives that enable access for deaf and hard of hearing users. The chapter will conclude with a brief overview of technologies that have been developed to assist with communication.

40.2 HEARING LOSS

According to the National Institute on Deafness and Other Communication Disorders (NIDCD), about 28 million people in the United States have some degree of hearing loss (National Institutes of Health [NIH] [2004]) (see also, Mitchell [2005], [2006]). This is a sizable population to be considered in the design of computer interfaces. Numbers alone, however, obscure some significant differences among the individuals who experience hearing loss. Degree and type of loss, age of onset of loss, as well as family, educational, and societal influences will all contribute to the experience and abilities of an individual who has a functional hearing loss.

Some individuals will have relatively little hearing loss, while others will experience a profound loss. Individuals who are hard of hearing will generally have some hearing. The ability that an individual user will have to make use of their residual hearing, however, is not a straightforward calculation of decibel (dB) loss. People with hearing loss will experience difficulty with pitch, timbre, and loudness, but, critically, will also experience difficulty with speech perception. Factors such as type of loss (e.g., conductive, sensorineural, mixed, or central) will have a major effect on the user experience. People with sensorineural hearing loss (such as resulting from lengthy exposure to loud noises or as a result of aging) generally will have more difficulty perceiving speech than people with conductive hearing losses that result from difficulties in middle ear functioning. The extent to which an individual makes use of this hearing for communication and whether the individual can hear computer sounds, however, varies greatly.

In addition, the way in which an individual having a hearing loss interacts with their environment may also be influenced by societal factors. Whether individuals identify themselves as Deaf (with an uppercase "D" as a member of the Deaf Community), deaf, or hard of hearing is indicative of cultural identity (Lane 1992; Padden and Humphries 1988, 2005).* Membership in the Deaf Community is determined more by shared language and worldviews rather than by results of audiometric tests. For example, people may lose their hearing with age. These people, while deaf by the audiometric definition, would not share the culture of the Deaf Community.

In the United States, the language of the Deaf Community is American Sign Language (ASL). Other countries and locales have their own native sign languages shared by members of Deaf Communities in those areas. Interestingly, these signed languages are not based on the spoken languages of the region. People are often surprised to learn that ASL is more similar to French Sign Language, from which it originated (Lane 1984), than it is to British Sign Language. Beginning with the seminal work of Ursula Bellugi and colleagues in the 1970s, linguists, cognitive psychologists, and brain researchers have studied native sign languages and their users for clues as to the origin of language and the biological nature of language (Emmorey and Lane 2000; Klima and Bellugi 1979; Erard 2005).

Cochlear implants are a medical intervention that has received much attention in the last couple of decades. The decision as to whether or not to have a cochlear implant is often a complex one, as was explored in the movie *Sound and Fury* (Aronson 2000; see also Hyde and Power [2006]). From the standpoint of a user, an implant is not the same as perfect hearing, but does allow the user to hear sounds and, with training, may greatly aid in the perception of speech (Chorost 2005).

40.3 HEARING AND LANGUAGE

Some, but not all, deaf and hard of hearing individuals use sign language. The type of signed language depends on the user's life experiences. Deaf children born to deaf parents, regardless of severity of hearing loss, will generally acquire a sign language, such as ASL, natively as hearing children of hearing parents acquire spoken language. People who lose their hearing late in life generally will not master a sign language. In between, there are many variations. Deaf signers may be exposed to a native sign language as adults and as a result, may only acquire partial mastery (Newport 1990). Other deaf signers will be exposed primarily to manual forms of English (or other spoken languages), rather than natural sign languages. Many schools that use sign language in the classroom do not use a natural sign language, but rather a representation of the spoken language that is signed. Forms of signed English borrow signs from ASL, but these signs are produced in English word order rather than using ASL sentence structures (for an extended discussion, see Lane, Hoffmeister, and Bahan [1996]). Children attending schools that use some form of signed English may not be exposed to ASL. Moreover, schools that educate students using an oral approach, in which speech is the primary means of communication in the classroom, also may not be exposed to ASL.

A person who is profoundly deaf from birth may have difficulty acquiring mastery of the spoken language, be it presented auditorily or in print. It is not surprising that someone who has never heard speech will have difficulty perceiving or producing it. In a large-scale study of deaf and hard of hearing children attending schools using an oral approach to education, Conrad (1979) reported that profoundly deaf children rarely acquired sufficient lipreading skills to allow easy participation in conversations. He found, on average, that these children (with hearing loss greater than 85 dB) could only comprehend about 25% to 28% of the words through lipreading that they could comprehend through reading. Even among these orally educated students, fewer than 20% had speech that was rated even fairly easy to understand. While the statistics are better for hard of hearing children, Conrad found that even these students (with hearing loss less than 65 dB) could only comprehend about 36% of the words through lipreading that they could comprehend through reading. Nearly 85% of these hard of hearing students, however, had speech that was rated at least fairly easy to understand.

Perhaps more surprising may be the fact that many deaf and hard of hearing individuals have difficulty with reading (Conrad 1979; Gallaudet Research Institute 2003). To understand this, it is necessary to realize that reading is based on the underlying spoken language. Spoken languages are composed of sounds, linguistically defined as "phonemes." These phonemes correspond, albeit not always in a one-to-one relationship in English, to letters or letter combinations. This is true of all alphabetic languages. Learning to read is generally considered to be learning to map the print onto the spoken language the person already knows (Brady and Shankweiler 1991). In addition, speech plays a critical role in the short-term memory processes that serve understanding of grammar and text comprehension (Lichtenstein 1998). In the case of deaf readers, however, it cannot be taken for granted that reading will build on a firm understanding of the structure of the spoken language, but there are no absolutes. Some prelingually, profoundly deaf children become excellent readers, while some with lesser degrees of hearing loss do not (Conrad 1979; Hanson 1989).

Interestingly, research has recently turned to an examination of signed languages as an influencer in the development of skilled reading. It has long been known that deaf children of deaf parents, on average, acquire greater mastery of

* Terms such as hearing-impaired and deaf-mute are generally considered to have negative connotations. For a discussion of this, see "What Is Wrong with the Use of These Terms: 'Deaf-mute,' 'Deaf and dumb,' or 'Hearing-impaired'?" by the National Association of the Deaf (available at http://www.nad.org/site/pp.asp?c5foINKQMBF&b5103786). The style manual of the American Psychological Association (APA) recommends use of nondiscriminatory language in all publications (see http://www.apastyle.org/disabilities.html).

reading and writing than deaf children of hearing parents. Is this due to early exposure to sign language? While a number of both intellectual and societal factors have been considered as contributors to this disparity, interest has focused on the issue of language. Evidence is now emerging as to the important role that early mastery of a sign language can have on second language learning for deaf students as they acquire reading and writing skills (Padden and Hanson 2000; Padden and Ramsey 2000).

In short, for any deaf or hard of hearing individual, language experience cannot be assumed. That individual may or may not sign, speak clearly or lipread well, or have reading skills consistent with those of the hearing population. This knowledge has implications for designers who seek to address the needs of deaf and hard of hearing users.

40.4 DISPLAY TECHNOLOGIES

In many ways, computers and other technologies have proven to be of great benefit to deaf and hard of hearing users. The largely visual nature of information on the Internet makes this information accessible to deaf and hard of hearing users. Instant messaging and e-mail facilitate communication with deaf and hearing family, friends, and coworkers, while network-connected wireless devices such as PDAs and pagers are seen as lifelines that can be used in place of cell phones. As would be suspected from the previous discussion, however, effective interfaces for these and other technologies for use by deaf and hard of hearing individuals must take into account both sensory and language considerations. In particular, the increasing reliance on sound and speech interfaces to convey information can have serious consequences for individuals who have a hearing loss.

40.4.1 Audible Signals

Sounds have become increasingly popular in computer interfaces. They have long been used to convey information about new messages and have become popular as problem alerts, such as when an error has been committed. These sound events are considered attention-grabbing events for users whose visual attention may otherwise be engaged. For any user who has a hearing loss, however, sounds will be a problem.

A number of considerations can help provide the necessary visual support for a user who is deaf or hard of hearing (Vanderheiden 1994). It is necessary to provide visual forms for all auditory information. Critically, these visual cues should be sufficiently noticeable so that they catch the attention of a person who may not be looking directly at the computer screen. Operating systems have features that can provide such visual alerts. For example, Windows® has accessibility features that allow users to set up their systems to have captions or visual warnings displayed for sound events. These are helpful, although they may not give the full range of information carried by a sound event. For example, the meaning of a sound event may differ based on the tone of the signal or when the sound is produced. While visual captions and warnings may alert a deaf user that a sound event has occurred, they will be unable to convey these more subtle distinctions. It is important for designers to give careful consideration to sound events to ensure that crucial information is available by a nonauditory means for deaf and hard of hearing users.

Multimodal interfaces, as the name implies, are designed to support a range of perceptual capabilities. In theory, this would seem ideal for users who are deaf or hard of hearing as visual alternatives to auditory materials should be available. Multimodal interfaces, however, do to always present all information on both modalities. The emphasis in many multimodal interfaces is representing information via speech that would otherwise be conveyed by print or some other visual means. To the detriment of deaf and hard of hearing users, often less attention is given to ensuring that all auditory material be visually conveyed as well.

40.4.2 Multimedia Interfaces

Multimedia uses a combination of text, sound, pictures, animation, and video to present information. Traditionally, games and educational software have exploited the richness of multimedia, but the advent of high-speed Internet communications has enabled the use of multimedia for a number of engaging applications on the web. For the present discussion, consideration will be given to multimedia presentations as they may impact access for deaf and hard of hearing users.

Multimedia material is inherently sensory. The technology offers eye-catching visual displays and attention-grabbing sound effects. To the extent that the information conveyed visually and auditorily is the same, information can be reinforced for users who have both channels available to them. To the extent that different information is presented in the two channels, however, users who have a functional loss of one of the channels will not have full access to that information. As it relates to the present discussion, this means that any information that is carried solely by sound or speech will be unavailable to deaf and hard of hearing users. Accessibility guidelines for multimedia products and web pages require equivalent visual presentations (e.g., see Brewer and Dardailler [1999]; U.S. General Services Administration [n.d.]).

Consider the growing popularity of video on the web. Video material is an extremely effective way to convey information and younger generations of computer users have come to expect video to be part of their computer experience. For deaf and hard of hearing users, the voice-over that is common in video will be inaccessible. Words spoken by persons in view of the camera also are largely inaccessible. Even people skilled at lipreading cannot lipread video conversations that have poor lighting or poor resolution, or speakers who turn away from the camera. Additionally, the video may present music or sound events (e.g., doorbells or animals noises) that contribute significantly to events on the video. These, too, are unavailable to deaf and hard of hearing users. Technologies such as captioning and sign-language translation exist to provide alternative presentations of sound events and speech. Designers wishing to make their applications universally accessible should consider these alternatives and incorporate them into their applications.

40.4.2.1 Captioning

Captioning provides a print alternative to speech and sound events. It is much like subtitling, except that it is specifically designed for deaf and hard of hearing users and, thus, will include comments in the captioning about sounds (e.g., "<music playing>" or "<sounds of child crying>") that may not be included in subtitles of foreign language videos. Captioning of certain television programming is mandated in the United States by the Federal Communications Commission and is considered to be beneficial to a large number of users, not only those with hearing loss. It has been shown, for example, to improve reading abilities of children and to benefit second language learners. Designers who use multimedia materials have an obligation to their full audience to provide captioning of audio and video materials. The listing of a number of resources for captioning software is available at the Closed Captioning website (n.d.).

As might be anticipated from the previous review of reading levels of deaf and hard of hearing users, there has been some controversy about what language level should be used for captioning. Simply put, the issue revolves around the question of whether captions should be verbatim transcripts or simplified captioning should be provided. Verbatim captioning is generally preferred by users themselves (NIH 2002) and is often cited as having the potential to improve reading skills (Steinfeld 2001).

40.4.2.2 Signing

For sign language users, there are sign language alternatives to captioning. These sign interfaces have been defined as ways of representing signed languages on a computer such that signing can be stored, displayed, and manipulated to facilitate computer interaction (Frishberg et al. 1993). It should be noted that these interfaces have often been employed not only for making audio and speech materials accessible to deaf and hard of hearing users, but they have also been used for language learning by both deaf and hearing people. A number of software applications have been developed that use sign language as a means of teaching reading skills and writing to deaf signers or teaching sign language skills to individuals wishing to learn to sign.

Importantly, software applications that purport to use "sign language" or ASL differ in significant ways in the language that is being used. Notably, many of these applications use fingerspelling. Fingerspelling is not a natural signed language, but rather is a derivative of print. Specifically, in fingerspelling, there is one handshape for each letter of the alphabet and words are spelled out letter by letter on the hands. Thus, the word "language" would be spelled out by eight distinct handshapes spelling L-A-N-G-U-A-G-E. Other sign language interfaces use a form of signed English, rather than the native sign language. For young children and others not fluent in English, interfaces that use fingerspelling or signed English transliterations of text may not meet their needs.

In many cases, the goal when using sign interfaces is to provide ASL translations. Given the present state of the art, automatic translation from English to ASL is not possible; however, many current efforts are directed at facilitating this translation, acknowledging the need for translation into ASL. In what follows, a few of the options for sign language presentation of audio and multimedia will be discussed.

Ideal in many respects would be to have a live person signing an ASL version of print and multimedia materials. Hanson and Padden (1989, 1990) used videodisc technology in the earliest computer-based attempt at a bilingual ASL/English approach to reading and writing instruction for signing children. The work combined ASL video and the translated English text on one screen. In this and other language learning situations where the users are young children still developing both ASL and English skills, this use of live signing has been particularly effective (Frishberg et al. 1993).

The advent of high-speed networks and Internet video has created opportunities for web applications that use live signing interfaces. For example, classroom applications have been developed (e.g., see King [2000]; Laurent Clerc National Deaf Education Center [n.d.]; for a demo, see "Sample Web Page," Gallaudet University [n.d.]). Additionally, video blogs have created the opportunity for blogs to be signed by the blogger, rather than written (Lamberton 2005), and video e-mail allows signers to communicate through signing by creating video recordings to be transmitted as e-mail messages (e.g., see Road Runner Video Mail [n.d.]; also, Vibe Video Mail [n.d.]).

Although ideal from the language perspective, other constraints may argue against the use of live signing in an interface. Lack of access to high-speed networks for video transmission may be an issue, but often the problem is the desire for automatic translation of software and web content that live signing does not provide. Live signing requires the prior recording of the signed material. Once recorded, changes to the video require a new recording. Because of this, live signing is not practical in applications that require that sign versions of audio or multimedia be created in real time. Short of having an interpreter doing the translations, live signing is not possible in these situations.

One approach to automatic sign presentation is what might be called "concatenated signing" (e.g., see iCommunicator [n.d.]; Signtel Inc. [n.d.]). With this, a software program is used to create word strings or sentences by concatenating signs produced by a live signer. The program starts with a vocabulary of stored individual signs. These signs can be strung together to form phrases and sentences. To prevent a jerky appearance that would occur through the simple

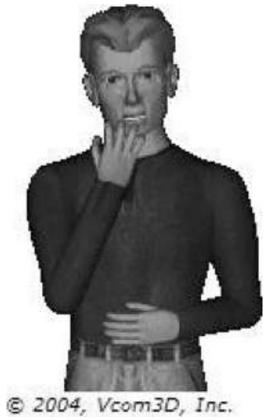


FIGURE 40.1 An American Sign Language sign avatar. The full animation of this avatar can be viewed on http://www.vcom3d.com/ASL.htm. © Vcom3D, Inc., 2004. All rights reserved.

production of a list of signs, algorithms are used to smooth the transitions between these signs. While these transitions are not as natural as live signing, the signing is legible and suggests an interesting approach to sign language interfaces.

A technique that has generated much interest in recent years is signing avatars (e.g., see Cox et al. [2002]; Karpouzis et al. [in press]; Kennaway [2002]; Sims [2004]). These avatars use virtual-reality techniques to produce animated signing. The specific techniques differ for the various avatars, but central to all is that they display computer generated signing. Some systems are able to display not just hands, but full signers, so that facial expressions as well as hand movements are shown. See, for example, the SigningAvatar® shown in Figure 40.1 that illustrates the sophistication of these animations.

Signing avatars have been used in education and have potential for applications such as translation of web pages, television programs, and conversational dialog. Chief among the virtues is that, unlike natural (live) sign language applications that are limited to prerecorded materials, avatars can generate signed versions of English words "on the fly." They also have the advantage of not requiring large downloads for web usage.

While the avatars can sign ASL when preprogrammed, the language translation work needed for automatic translation into ASL (or other signed language) is not ready to support this rendering on the fly. A current research focus in avatar work is on natural language translation and exploring techniques to display native sign languages by avatars (e.g., see Huenerfauth [2005]).

40.5 INPUT TECHNOLOGIES

Standard keyboard and mouse input technologies offer no barriers to users who have a hearing loss. The emergence of new interfaces, however, presents alternatives that will impact the way in which we are able to interact with computers. We consider here both speech and gestural interfaces as they may influence interactions for deaf and hard of hearing users.

40.5.1 SPEECH INTERFACES

Conversational speech interfaces have appeal as a natural means of interacting with computers. These interactions can range from simple, even one-word commands to full dictation of documents and user collaborations. For deaf and hard of hearing individuals, such interactions require consideration of user needs. First, these interactions can involve speech output which, as already discussed, requires a visual display alternative. Second are problems with speech input. Some deaf and hard of hearing individuals utilize speech and would be interested in taking advantage of speech input. Speech recognition, however, may be more problematic for these speakers than for hearing speakers.

As mentioned briefly in a previous paragraph, the speech of deaf and hard of hearing individuals is not always highly intelligible to hearing listeners. Since speech recognition engines can be trained to the voice and pronunciations of an individual speaker, however, couldn't a recognizer be trained to understand the speech of an individual deaf speaker, even if the speech is not completely intelligible to listeners? The difficulty is that recognizers require consistent speech. Research has shown that the speech of deaf and hard of hearing speakers often is more variable than the speech of hearing speakers (McGarr 1987; McGarr and Lofqvist 1988). For example, there is more acoustic variation in the pronunciation of a single phoneme by a deaf or hard of hearing speaker than there is in the pronunciation of that phoneme by a hearing speaker. Hearing listeners are very tolerant of this variability; speech recognizers are less so. Thus, deaf and hard of hearing users who many wish to use their speech for input may well find recognition less accurate than it is for hearing speakers.

Speech interfaces are often seen as useful alternatives to visual interfaces in situations when computers or keyboards are not available. For hearing users, this alternative of a speech interface may be highly desirable. Many deaf and hard of hearing users, however, will not be interested in speech interfaces; others may experience difficulties in using them. Although situational demands may sometimes dictate the use of conversational interfaces, care must be taken that outside of the situational context there exists a means for deaf and hard of hearing users to access the same information that hearing users access.

40.5.2 SIGN INTERFACES

Designed specifically for deaf and hard of hearing users, sign recognition is a specific and complex subset of gesture recognition technologies having the goal of automatically converting signed language to text or speech. These technologies have been investigated for a number of years, primarily to address the need to facilitate communication between signers and nonsigners. They also have the potential to provide an alternative means of natural language input to computers.

The task of recognizing full ASL or other natural sign languages is a difficult problem. The first reason is that an individual sign varies depending upon its context. Take, for example, the sign for the word GIVE. This sign has a specific shape combined with a variable movement. The movement reflects who is giving and to whom. I GIVE-TO YOU, YOU GIVE-TO ME, I GIVE-TO HIM, HE GIVES-TO ME, and I GIVE-TO ALL-OF-YOU each has a different movement that indicates subject and object. A second reason for difficulty in sign-language recognition is that several pieces of linguistic information are produced in parallel. For example, facial expression carries critical grammatical information. Thus, a full language recognizer needs to recognize not only the hand gestures of ASL, but must also recognize certain facial information relevant to the grammar. Such facial elements include eyebrow position, eye gaze, and mouth movements.

As with interfaces that produce signs, many systems that purport to perform sign-language recognition deal with recognition of fingerspelling handshapes rather than recognition of ASL sentences or even ASL signs. Using fingerspelling handshapes rather than ASL signs certainly constrains the size of the problem. Because there are only 26 handshapes in the English alphabet, this represents a much more manageable problem space than the recognition of full ASL signing. While it presents an alternative to keyboard typing of words, it doesn't provide the type of natural language interaction with computers that is afforded by speech interfaces.

Technologies for recognizing signs have tended to use either instruments worn by the signer or computer vision techniques. The first of these approaches has the signer wear a specially designed glove or sensors placed on their joints that allow a computer to track movement (e.g., see Braffort [1996]; Fang, Gao, and Zhao [2003]; Hernandez-Rebollar, Lindeman, and Kyriakopoulos [2002]; Kadous [1996]; Wang, Gao, and Ma [2002]). In contrast, computer vision techniques use cameras to provide input to a computer about a signer's movements and facial gestures (e.g., see Brashear et al. [2003]; Kadous [1996]; Lee et al. [2005]; Vogler and Metaxas [2001]). These inputs are then analyzed using a variety of techniques such as neural networks or Hidden Markov Models that then recognize the sign.

Critically, however, this recognition does not do language translation. Thus, the output will be a one-to-one mapping of a sign into print or speech. As mentioned with the avatar work presented earlier, the ASL/English rules that would be needed for such translation are not developed to a state where automatic translation can occur; however, advances are being made. Recently, for example, Hernandez-Rebollar (2005) presented work designed to translate signed input into English phrases. In that work, the translation was enabled by having a limited number of phrases that the system could recognize, thus constraining the problem space.

The current state of the art for sign recognition is not as advanced as speech recognition. As researchers continue to work on the problem, however, advances can be expected. For signers, it might be the case that conversational sign interactions will one day be possible, much as speech recognition now allows speakers to benefit from conversational speech interactions.

40.6 TECHNOLOGY AND COMMUNICATION

Deaf and hard of hearing individuals have difficulties not only with computer interfaces, but also experience significant difficulties in certain communication situations. Telephone conversations, as well as one-on-one conversations, group discussions, and presentations or classroom lectures, are all problematic. No discussion of HCI for deaf and hard of hearing individuals would be complete without at least a brief mention of technologies that have been developed to aid in these situations.

Telephones have long been a source of difficulty for deaf and hard of hearing people. Alexander Graham Bell was a teacher of deaf students and was married to a woman who was deaf. His interest in finding improved ways to communicate with deaf speakers led to his invention of the telephone. Ironically, however, over the years, telephones created a number of barriers for deaf and hard of hearing individuals in the workplace and other situations. To overcome these barriers, a number of assistive devices have been developed (Lazzaro 1993). Amplification and adapters exist for many phones that will allow hard of hearing individuals or people with cochlear implants to hear phone conversations. Teletype devices (TTYs and TDDs) as well as some computer applications allow deaf users to type conversations that are carried over phone lines.

Operator-assisted relay services have been established to enable conversations between deaf and hearing individuals. The relay personnel serve as a bridge between the two conversation participants, translating typed information into speech for the hearing participant and translating speech into written text for the deaf participant. Hard of hearing speakers can speak directly with others using a captioning service that provides, nearly in real time, a printed transcript of the conversation to support hard of hearing users (CapTel 2005). Various means of enabling signed conversations over the telephone have been explored over the years, but the advent of video phones has now made feasible the option of signed phone conversations.

The current prevalence of conference calls in the workplace has created a new set of problems for deaf and hard of hearing workers. Similarly, classroom situations or lecture presentations also create significant difficulties for deaf and hard of hearing attendees. Even individuals skilled in lipreading have difficulty in these situations because the speaker's face is rarely visible with sufficient resolution for lipreading. Sign language interpreting and captioning are two means of providing accommodation for participants unable to hear or understand the speech in these situations.

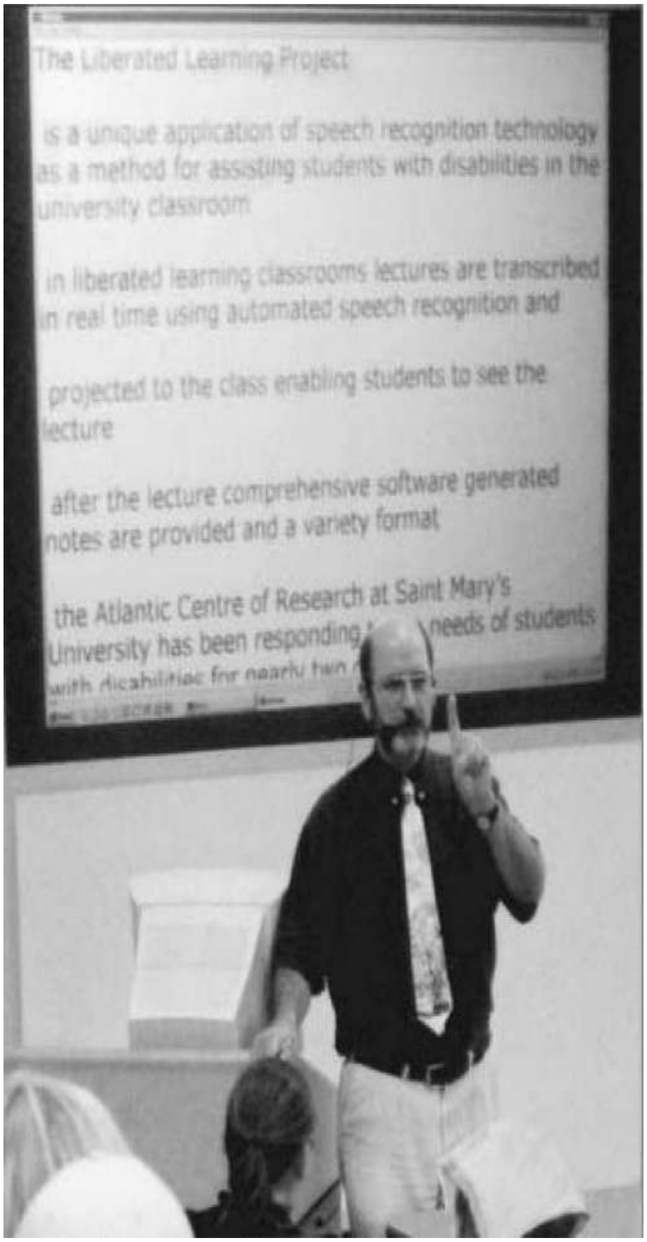
Sign language interpreters allow signers to participate in meetings by providing real-time translation of the spoken conversation into sign. This is a two-way translation service, such that the signing participant is also able to participate by having the interpreter speak their signed utterances. Remote interpreting is a technology that addresses the problem of a shortage of skilled interpreters, particularly in some locations. The interesting aspect of this is that the interpreter need not be present at the location of any of the conference participants. The deaf or hard of hearing participant views the remotely located interpreter on a computer display or TV screen. The interpreter listens to hearing participants by telephone and provides sign-language interpreting for the deaf participant. The interpreter also voices what the deaf person signs for hearing participants. This remote interpreting can be used similarly to the telephone relay service, with participants in different locations (Video Relay Service, or VRS), or in situations where the participants are located in the same room (Video Relay Interpreting, or VRI).

Communication Access Real-time Translation (CART) provides real time captioning, enabling discussions and presentations to be transcribed into text for deaf and hard of hearing participants. Typically, this is done by a person who creates the captions. As with sign language interpreting, the captioner can be physically present or remotely located, listening to the conversation by phone and transmitting the text via a network to a computer screen or projected display. The deaf participant types their questions or comments. That input is then read aloud for the other participants, either by a captioner or using computer text-to-speech technology (e.g., see Caption First [n.d.]; Viable Technologies [n.d.])

The ability to automatically transcribe speech was envisioned more than 100 years ago as a technology that held great promise for deaf and hard of hearing people, even though such technology was hardly imaginable at the time (Fay 1883*)*. As speech-recognition systems have matured over the last quarter century, a number of applications designed for deaf users have been explored (Bain et al. 2005; Stinson and Stuckless 1998).

The Liberated Learning Project is one example of using automatic speech recognition in the classroom (Bain, Basson, and Wald 2002). In this effort, the classroom teacher speaks into a microphone that transmits his or her speech to a computer to perform the recognition. The transcript of the lecture is displayed in close to real time on a screen at the front of the classroom. Shown in Figure 40.2 is a classroom situation for the Liberated Learning Project.

While recognition technologies have improved over the years, they still do not attain 100% correct performance for dictation. To provide students with accurate transcripts for lecture notes, the professors who participate in the Liberated Learning Project review the transcript for inaccuracies and make corrections after a lecture is completed. The corrected transcripts are then made available to students (Bain, Basson, and Wald 2002). Other efforts that use automatic speech recognition for captioning use different methods for correcting errors in transcripts. These other methods include having a trained speaker "shadow" the speech to produce more reliable recognition, and/or having a person correcting errors in real time (e.g., see Bain et al. [2005]; Robson [2001]; Viable Technologies [n.d.]).



The Liberated Learning Project

is a unique application of speech recognition technology as a method for assisting students with disabilities in the university classroom

in liberated learning classrooms lectures are transcribed in real time using automated speech recognition and

projected to the class enabling students to see the lecture

after the lecture comprehensive software generated notes are provided and a variety format

the Atlantic Centre of Research at Saint Mary's University has been responding needs of students with disabilities for nearly two

FIGURE 40.2 An example classroom lecture transcribed in real time. From Bain et al. (2005); republished with permission.

40.7 CONCLUSIONS

Computer technologies are very important in the lives of many deaf and hard of hearing people. These technologies can ease communication between coworkers, friends, family members, neighbors, and a variety of services. These technologies have also provided access for deaf students, employees, and individuals to have full access to rich multimedia services, shopping, news, and general information. In short, barriers long encountered by deaf and hard of hearing people are now being overcome through the use of technology.

Deaf and hard of hearing users represent a population that, in itself, is diverse in both hearing and language experiences and skills. Applications that offer flexibility of language (e.g., captioning, signed English, or ASL) will be most accessible to users who have a hearing loss. At a minimum, however, developers and designers need to ensure that information is not carried by the auditory channel alone. Paramount is the need to ensure that any audible information, be it a sound alert, speech prompt, or other auditory event, has a visual counterpart. Such considerations are not only good design, but are also mandated by a growing number of regulations worldwide. The ability of everyone to participate in our increasingly technological society is crucial.

ACKNOWLEDGMENTS

I'd like to thank Al Noll, Debra Noll, and John Richards for valuable comments on early drafts of this chapter.

REFERENCES

Aronson, J. 2000. *Sound and Fury* [Motion picture]. United States: New Video Group.

Bain, K., S. Basson, A. Faisman, and D. Kanevsky. 2005. Accessibility, transcription, and access everywhere. *IBM Syst J* 44(3): 589–605.

Bain, K., S. Basson, and M. Wald. 2002. Speech recognition in university classrooms: Liberated learning project. In *Proceedings of the 5th International ACM SIGCAPH Conference on Assistive Technologies* (*ASSETS'02*), 192–6. New York: ACM.

Brady, S., and D. Shankweiler. 1991. *Phonological Processes in Literacy: A Tribute to Isabelle Y. Liberman*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Braffort, A. 1996. A gesture recognition architecture for sign language. In *Proceedings of the 2nd International ACM SIGCAPH Conference on Assistive Technologies* (ASSETS'96), 102–9, Vancouver, British Columbia, Canada. New York: ACM.

Brashear, H., T. Starner, P. Lukowicz, and H. Junker. 2003. Using multiple sensors for mobile sign language recognition. In *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, 45–52, White Plains, NY. Washington, DC: IEEE Computer Society.

Brewer, J., and D. Dardailler. 1999. Web content accessibility guidelines 1.0. University of Wisconsin-Madison, Trace Research and Development Center. http://www.w3.org/TR/WAI-WEBCONTENT/ (accessed December 7, 2005).

CapTel. (n.d.). Introducing the captioned Telephone. http://www.captionedtelephone.com/aboutcaptel.phtml (accessed December 7, 2005).

Caption First. (n.d.). Overview of our services. http://www.captionfirst.com/overview.htm (accessed January 20, 2006).

Chorost, M. 2005. *Rebuilt: How Becoming Part Computer Made Me More Human.* New York, NY: Houghton Mifflin Company.

Closed Captioning Web. (n.d.) Closed captioning web: Software links. http://www.captions.org/softlinks.cfm (accessed December 7, 2005).

Conrad, R. 1979. *The Deaf Schoolchild.* London: Harper & Row.

Cox, S., M. Lincoln, J. Tryggvason, M. Nakisa, M. Wells, M. Tutt, and S. Abbott. 2002. TESSA, a system to aid communication with deaf people. In *Proceedings of the 5th International ACM SIGCAPH Conference on Assistive Technologies* (*ASSETS'02*), 205–12. New York: ACM.

Emmorey, K., and H. Lane. 2000. *The Signs of Language Revisited: An Anthology to Honor Ursula Bellugi and Edward Klima*. Mahwah, NJ: Lawrence Erlbaum Associates.

Erard, M. 2005. The birth of a language. *New Sci* 188(2522):46–9.

Fang, G., W. Gao, and D. Zhao. 2003. Large vocabulary sign language recognition based on hierarchical decision trees. In *Proceedings of the 5th International Conference on Multimodal Interfaces*, 125–31. New York: ACM.

Fay, E. A. 1883. The glossograph. *Am Ann Deaf* 28:67–9.

Frishberg, N., S. Corazza, L. Day, S. Wilcox, and R. Schulmeister. 1993. Sign language interfaces. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 194–7. New York: ACM.

Gallaudet Research Institute. 2003. Literacy and deaf students. http://gri.gallaudet.edu/Literacy/ (accessed December 7, 2005).

Gallaudet University. (n.d.). Sample web page with embedded real media video. http://academic.gallaudet.edu/pages/iced2000/real/stellaluna_smil.html (accessed January 20, 2006).

Hanson, V. L. 1989. Phonology and reading: Evidence from profoundly deaf readers. In *Phonology and Reading Disability: Solving the Reading Puzzle*, ed. D. Shankweiler and I. Y. Liberman, 69–89. Ann Arbor, MI: University of Michigan Press.

Hanson, V. L., and C. A. Padden. 1989. The use of interactive video for bilingual ASL/English instruction of deaf children. *Am Ann Deaf* 134:209–13.

Hanson, V. L., and C. A. Padden. 1990. Bilingual ASL/English instruction of deaf children. In *Cognition, Education, and Multimedia: Exploring Ideas in High-Technology*, ed. D. Nix and R. Spiro, 49–63. Hillsdale, NJ: Lawrence Erlbaum Associates.

Hernandez-Rebollar, J. L. 2005. Gesture-drive American Sign Language phraselator. In *Proceedings of the 7th International ACM Conference on Multimodal Interfaces* (*ICMI'05*), 288– 92. New York: ACM Press.

Hernandez-Rebollar, J. L., R. W. Lindeman, and N. Kyriakopoulos. 2002. A multi class pattern recognition system for practical fingerspelling translation. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces* (ICMI'05), 185–90. Washington, DC: IEEE Computer Society.

Huenerfauth, M. 2005. Representing coordination and noncoordination in an American Sign Language animation. In *Proceedings of the 7th International ACM SIGACCESS Conference on Assistive Technologies* (*ASSETS'05*), 44–51. New York: ACM Press.

Hyde, M., and D. Power. 2006. Some ethical dimensions of cochlear implantation for deaf children and their families. *J Deaf Stud Deaf Educ* 11(1):102–11.

iCommunicator. (n.d.). *iCommunicator.* http://www.myicommunicator. com/ (accessed January 4, 2006).

Kadous, M. W. 1996. Machine recognition of Auslan signs using Powergloves: Towards large-lexicon recognition of sign languages. In *Proceedings of WIGLS, The Workshop on the Integration of Gestures in Language and Speech, Wilmington Delaware*, ed. L. Messing, 165–74.

Karpouzis, K., G. Caridakis, S.-E. Fotinea, and E. Efthimiou. 2007. Educational resources and implementation of a greek sign language synthesis architecture. *Computers & Education* 49(1):54–74. doi:10.1016/j.compedu.2005.06.004.

Kennaway, R. 2002. Synthetic animation of deaf signing gestures. In *Gesture and Sign Language in Human-Computer Interaction: International Gesture Workshop (GW 2001, LNAI 2298)*, 146–57.

King, C. 2000. *Online Learning at Gallaudet University.* Gallaudet University. http://academic.gallaudet.edu/pages/iced2000/iced2000_GUonlinelearning.PDF (accessed January 20, 2006).

Klima, E. S., and U. Bellugi. 1979. *The Signs of Language*. Cambridge, MA: Harvard University Press.

Lamberton, J. 2005. Jason Lamberton's video blog. http://video.google.com/videoplay?docid56012463606293405795&q5gallaudet1university (accessed January 20, 2006).

Lane, H. 1984. *When the Mind Hears: A History of the Deaf.* New York: Random House.

Lane, H. 1992. *The Mask of Benevolence: Disabling the Deaf Community.* New York: Alfred A. Knopf.

Lane, H., R. Hoffmeister, and B. Bahan. 1996. *A Journey into the Deaf-world*. San Diego, CA: Dawn Sign Press.

Laurent Clerc National Deaf Education Center, Gallaudet University. (n.d.). Shared reading project: Chapter by chapter—The Thinking Reader. http://clerccenter.gallaudet.edu/Literacy/programs/chapter.html (accessed January 20, 2006).

Lazzaro, J. J. 1993. *Adaptive Technologies for Learning and Work Environments.* Chicago, IL: American Library Association.

Lee, S., V. Henderson, H. Hamilton, T. Starner, H. Brashear, and S. Hamilton. 2005. A gesture-based American Sign Language game for deaf children. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (*CHI '05*), 1589–92. New York: ACM Press.

Lichtenstein, E. H. 1998. The relationships between reading processes and English skills of deaf college students. *J Deaf Stud Deaf Educ* 3(2):80–134.

McGarr, N. S. 1987. Communication skills of hearing-impaired children in schools for the deaf. *ASHA Monogr* (26):91–107.

McGarr, N. S., and A. Lofqvist. 1988. Laryngeal kinematics in voiceless obstruents produced by hearing-impaired speakers. *J Speech Hear Res* 31(2):234–9.

Mitchell, R. E. 2005. Can you tell me how many deaf people there are in the United States? Galludet Research Institute, Graduate School and Professional Programs. http://gri.gallaudet.edu/Demographics/deaf-US.php (accessed December 7, 2005).

Mitchell, R. E. 2006. How many deaf people are there in the United States? Estimates from the survey of income and program participation. *J Deaf Stud Deaf Educ* 11(1):112–9.

National Institutes of Health, National Institute on Deafness and Other Communication Disorders. 2002. *Captions for Deaf and Hard of Hearing Viewers*. NIH Publication No. 00-4834. http://www.nidcd.nih.gov/health/hearing/caption.asp#edit (accessed December 7, 2005).

National Institutes of Health, National Institute on Deafness and Other Communication Disorders. 2004. *Statistics About Hearing Disorders, Ear Infections, and Deafness*. http://www.nidcd.nih.gov/health/statistics/hearing.asp (accessed December 7, 2005).

Newport, E. 1990. Maturational constraints on language learning. *Cogn Sci* 14:11–28.

Padden, C. A., and V. L. Hanson. 2000. Search for the missing link: The development of skilled reading in deaf children. In *The Signs of Language Revisited: An Anthology to Honor Ursula Bellugi and Edward Klima,* ed. K. Emmorey and H. Lane, 435–47. Mahwah, NJ: Lawrence Erlbaum Associates.

Padden, C., and T. Humphries. 1988. *Deaf in America: Voices from a Culture*. Cambridge, MA: Harvard University Press.

Padden, C., and T. Humphries. 2005. *Inside Deaf Culture*. Cambridge, MA: Harvard University Press.

Padden, C., and C. Ramsey. 2000. American Sign Language and reading ability in deaf children. In *Language Acquisition by Eye*, ed. C. Chamberlain, J. Morford, and R. Mayberry, 165–89. Mahwah, NJ: Lawrence Erlbaum Associates.

Road runner video mail. (n.d.). http://vmail.vibephone.com/vm/vm_player?vmfile54dbc51d033b4cdbac583bf8ec3ef9cd3&vmpid51007 (accessed January 20, 2006).

Robson, G. 2001. Can realtime captioning be done using realtime voice recognition systems? http://www.robson.org/capfaq/online.html#VoiceRecognition (accessed December 7, 2005).

Signtel Inc. (n.d.). Signtel. http://www.signtelinc.com/main.htm (accessed December 7, 2005).

Sims, E. 2004. *Using Emerging Visualization Technologies to Provide Sign Language Access to the Web*. http://www.w3.org/WAI/RD/2003/12/Visualization/VCom3D.html (accessed January 20, 2006).

Steinfeld, A. 2001. The case for real time captioning in classrooms. http://www.cartinfo.org/steinfeld.html (accessed December 7, 2005).

Stinson, M., and R. Stuckless. 1998. Recent developments in speech-to-print transcription systems for deaf students. In *Issues Unresolved: New Perspectives on Language and Deaf Education*, ed. A. Weisel, 126–32. Washington, DC: Gallaudet University Press.

U.S. General Services Administration, Office of Governmentwide Policy, IT Accessibility & Workforce Division. (n.d.). *Section 508.* http://section508.gov/ (accessed January 20, 2006).

Vanderheiden, G. C. 1994. Application software design guidelines: Increasing the accessibility of application software for people with disabilities and older users. University of Wisconsin-Madison, Trace Research and Development Center. http://trace.wisc.edu/docs/software_guidelines/software.htm (accessed January 20, 2006).

Viable Technologies. (n.d.). Frequently asked Questions. http://www.viabletechnologies.com/faq.php (accessed December 7, 2005).

Vibe video mail. (n.d.). http://www.vibephone.com/vsg/htdocs/products/video-mail/index.jsp (accessed January 20, 2006).

Vogler, C., and D. Metaxas. 2001. Framework for recognizing the simultaneous aspects of American Sign Language. *Comput Vis Image Underst* 81:358–84. doi:10.1006/cviu.2000.0895, http://www.idealibrary.com

Wang, C., W. Gao, and J. Ma. 2002. A real-time large vocabulary recognition system for Chinese Sign Language. In *International Gesture Workshop, Springer Lecture Notes in Artificial Intelligence*, ed. I. Wachsmuth and T. Sowa, Vol. 2298, 86–95. New York, NY: Springer-Verlag.

Part VI

The Development Process

Section A: Requirements Specification

41 User Experience Requirements Analysis within the Usability Engineering Lifecycle

Deborah J. Mayhew, and Todd J. Follansbee

CONTENTS

41.1 INTRODUCING USER EXPERIENCE REQUIREMENTS ANALYSIS

As the World Wide Web has matured, the field of traditional software usability (which dates back to the late 1970s) has come to recognize—and integrate with—other qualities of what is now referred to as the web user experience (UX). As web capabilities have increased, graphic design has become a key quality of the UX. And in the case of e-Commerce websites, a relatively new quality of the UX design has emerged: persuasiveness. At this point, any e-Commerce designer or developer needs to recognize the importance of the following five different qualities of the total website UX:

1. Utility
2. Functional integrity
3. Usability
4. Persuasiveness
5. Graphic design

These qualities of the UX design are defined and discussed in Chapter 51. This chapter updates the corresponding chapter in the previous edition of this volume by expanding the topic of requirements analysis beyond traditional usability to address requirements relating to these other aspects of the total UX, particularly in the case of website design.

The following three key ingredients are necessary to ensure that an optimal UX design is achieved during software or website development:

1. Application of established design principles and guidelines
2. A structured methodology for design
3. Managerial and organizational techniques

At this point, well-established UX design principles and guidelines are available relating to usability (Tidwell 2010; Nielsen and Loranger 2006), persuasion (Eisenberg and Eisenberg 2005; Goldstein, Martin, and Cialdini 2008); and graphic design (Beaird 2010), based on objective research, and are reported in the literature. Many of these principles and guidelines are enumerated throughout different chapters in this book. Development organizations need to have access to professionals who are fluent in these design guidelines and who can participate in design efforts, so this general accumulated knowledge will find its way into their applications and websites.

Just having a design guru or two on board does not guarantee that appropriate design principles and guidelines will find their way into final websites or applications. Design is complex, and there is no simple cookbook approach to design that can rely on general principles and guidelines alone. Development organizations also need structured methods for achieving an optimal UX in their websites and applications.

Similarly, a well-structured and documented design methodology must be introduced and managed—it does not happen by itself. Thus, managerial and organizational techniques must be applied to ensure that the design methodology is followed and includes the application of well-established design principles.

Even when good management practices are being applied, either of the remaining two ingredients alone—design guidelines or design methods—is necessary, but not sufficient. Optimal UX design cannot be accomplished by the systematic application of generic guidelines alone, because every application or website and its intended set of users is unique. Design guidelines must be tailored for and validated against unique requirements, and this is what the structured methods accomplish. Conversely, applying structured methods without also drawing on well-established design principles and guidelines is inefficient at best and may simply fail at worst. Without the benefit of the initial guidance of sound design principles during first passes at design, a particular project with its limited resources may simply never stumble upon a design approach that works. For example, formal usability testing is a valuable and an objective method for uncovering usability problems. However, without a clear understanding of basic design principles and guidelines, as well as unique requirements data, solving the identified problems without incurring new ones will not be easy or likely.

Previous sections and chapters in this book refer to a broad variety of design principles and guidelines. Section A (Requirements Specification) in Part VI (The Development Process) of this volume addresses the need for methodology and provides chapters that address a variety of techniques that can be applied during the development process to achieve an optimized UX in their website or application. This chapter sets the stage for this section and, in particular, for the subsection on one stage of the development process: requirements analysis.

The *Usability Engineering Lifecycle* (Mayhew 1999) documented a structured and a systematic approach to address one aspect of UX design—usability—within the application or web development process. Here, we widen the scope from usability to UX, and change terminology as appropriate to reflect this change in scope.

The UX engineering lifecycle consists of a set of UX engineering tasks applied in a particular order at specified points in an overall development lifecycle.

Several types of tasks are included in the UX lifecycle, as follows:

- Structured requirements analysis tasks
- An explicit goal-setting task, driven directly from requirements analysis data
- Tasks supporting a structured, top–down approach to UX design that is driven directly from UX goals and other requirements data
- Objective evaluation tasks for iterating design toward UX goals

Figure 41.1 represents in summary, visual form, the original Usability Engineering Lifecycle (please note that in the figure, "OOSE" stands for Object Oriented Software Engineering). The overall lifecycle is cast in three phases: Requirements Analysis, Design/Testing/Development, and Installation. Specific tasks within each phase are presented in boxes, and arrows show the basic order in which the tasks should be carried out. Much of the sequencing of tasks is iterative, and the specific places where iterations would most typically occur are illustrated by arrows returning to earlier points in the lifecycle. Brief descriptions of each lifecycle task are given below, using the new term UX to widen the scope of tasks as described by Mayhew (1999).

41.1.1 Phase 1: Requirements Analysis

41.1.1.1 User Profile

A description of the specific user characteristics relevant to UX design (e.g., computer literacy, expected frequency of use, and level of job experience) is obtained for the intended website or application user population. This will drive tailored UX design decisions and identify major user categories for study in the contextual task analysis. This volume includes a set of chapters (Part V—Designing for Diversity) dedicated to exploring the unique needs and requirements of different types of user populations.

41.1.1.2 Contextual Task Analysis

A study of users' current tasks, workflow patterns, and conceptual frameworks is made, resulting in a description of current tasks and workflow, and an understanding and a specification of underlying user goals. These will be used to set UX goals and drive work reengineering (i.e., information architecture) and UX design. Section A (Requirements Specification) in Part VI (The Development Process) of this volume includes several chapters on topics related to task analysis.

41.1.1.3 UX Goal Setting

Specific qualitative goals reflecting UX requirements are developed, extracted from the user profile and contextual task analysis. In addition, quantitative goals (based on a subset of high-priority qualitative goals) are developed, defining minimal acceptable user performance and satisfaction

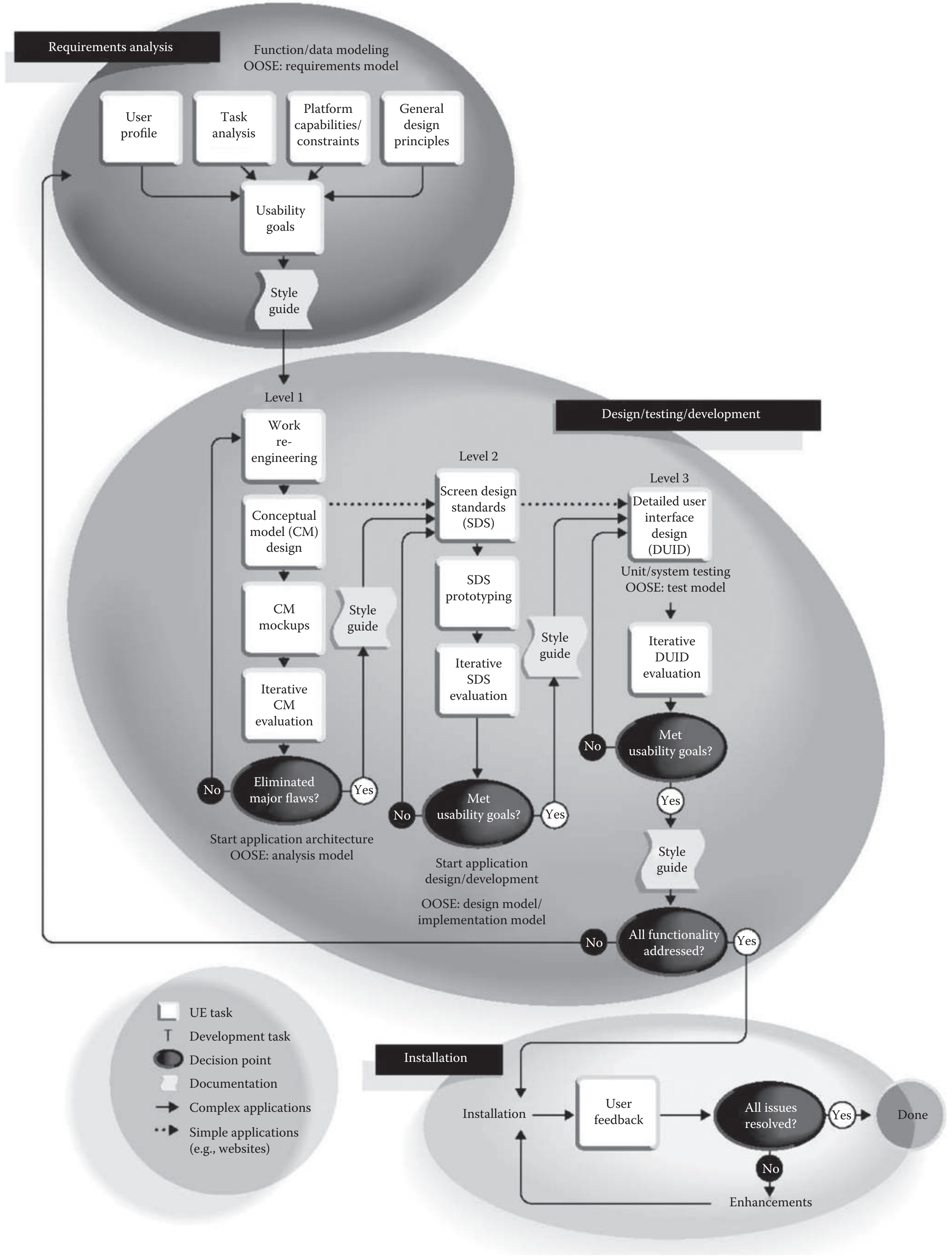


FIGURE 41.1 The Usability Engineering Lifecycle. (From Mayhew, D. J. 1999. *The Usability Engineering Lifecycle*. San Francisco, CA: Morgan Kaufmann Publishers.)

criteria (and in the case of e-Commerce, conversion rates). These UX goals focus later design efforts and form the basis for later iterative UX evaluation.

41.1.1.4 Platform Capabilities/Constraints

The user interface capabilities and constraints (e.g., screen size, screen resolution, browser brand and version, etc.) inherent in the technology platform chosen for the application (e.g., Apple Macintosh, MS Windows, and product-unique platforms) or website (e.g., browser brand and version) are determined and documented. These constraints will define the scope of possibilities for UX design. This volume includes chapters that describe platform capabilities that support many relatively new and innovative types of user interfaces, including multimedia (Chapter 17), adaptive interfaces and agents (Chapter 19), mobile devices and ubiquitous computing (Chapter 20), tangible interfaces (Chapter 21), and wearable computers (Chapter 12).

41.1.1.5 General Design Guidelines

Relevant general UX design guidelines available in the literature are gathered and reviewed. They will be applied in the future design process, along with all other project-specific information gathered in the previous tasks.

41.1.2 Phase 2: Design/Testing/Development

41.1.2.1 Level 1 Design

41.1.2.1.1 Work Reengineering

Based on all requirements analysis data and the UX goals extracted from them, user tasks are redesigned at the level of organization and workflow to streamline work and exploit changing software or web capabilities. No visual UX design is involved in this task, just abstract organization of functionality and workflow design. This task is sometimes referred to as “information architecture.”

41.1.2.1.2 Conceptual Model Design

Based on all the previous tasks, initial high-level design alternatives are generated. At this level, rules for the consistent presentation of navigational controls are established. Screen/page content design is not addressed at this design level.

41.1.2.1.3 Conceptual Model Mockups

Paper-and-pencil or prototype mockups of high-level design ideas generated in the previous task are prepared, representing ideas about high-level functional organization and conceptual model design (see Chapter 47 by Beaudouin-Lafon and MacKay in this volume for a discussion of prototyping tools and techniques). Detailed screen/page design and complete functional design are not in focus here.

41.1.2.1.4 Iterative Conceptual Model Evaluation

The mockups are evaluated and modified through iterative evaluation techniques such as formal usability and/or persuasion testing in which real, representative end users attempt to perform real, representative tasks with minimal training or intervention, imagining that the mockups are a real software or web user interface (see Section C: Testing and Evaluation, Part VI-The Development Process in this volume for chapters on testing and evaluation). This and the previous two tasks are conducted in iterative cycles until all major UX “bugs” are identified and engineered out of Level 1 (e.g., conceptual model) design. Once a conceptual model is relatively stable, system architecture design can commence.

41.1.2.2 Level 2 Design

41.1.2.2.1 Screen/Page Design Standards

A set of specific standards and conventions for all aspects of detailed screen/page design is developed, based on any industry and/or corporate standards that have been mandated (e.g., Microsoft Windows, Apple Macintosh, corporate website standards, etc.), the data generated in the Requirements Analysis phase, and the unique conceptual model design arrived at during Level 1 design.

41.1.2.2.2 Screen/Page Design Standards Prototyping

The screen/page design standards (as well as the conceptual model design) are applied to design the detailed UX to selected subsets of functionality. This design is implemented as a live prototype.

41.1.2.2.3 Iterative Screen/Page Design Standards Evaluation

An evaluation technique, such as formal usability and/or persuasion testing, is carried out on the screen/page design standards prototype, and then redesign/reevaluation iterations are performed to refine and validate a robust set of screen/page design standards. Iterations are continued until all major UX bugs are eliminated and UX goals seem within reach.

41.1.2.2.4 Style Guide Development

At the end of the design/evaluate iterations in Design Levels 1 and 2, we have a validated and a stabilized information architecture and conceptual model design, and also validated and stabilized set of standards and conventions for all aspects of detailed screen/page design. These are captured in a document called a style guide, which already documents the results of requirements analysis tasks. During detailed user interface design (see Section 41.1.2.3.1), following the conceptual model design and screen design standards in the style guide will ensure quality, coherence, and consistency—the foundations of a good UX.

41.1.2.3 Level 3 Design

41.1.2.3.1 Detailed UX Design

Detailed design of the complete software or website UX is carried out based on the refined and validated conceptual model and screen/page design standards documented in the style guide. This design then drives development.

41.1.2.3.2 Iterative Detailed User Interface Design Evaluation

Techniques such as formal usability and/or persuasion testing are continued during development to expand evaluation to previously unassessed subsets of functionality and categories of users and to continue to refine the UX and validate it against UX goals.

41.1.3 Phase 3: Installation

41.1.3.1 User Feedback

After a software application or website is installed and in production for some time, feedback is gathered to feed into design enhancements, design of new releases, and/or design of new but related products. The UX undergoes continuous testing and improvement throughout its life.

41.2 FOCUSING ON UX REQUIREMENTS ANALYSIS

This section (Section A-Requirements Specification, in Part VI-The Development Process) provides in-depth coverage of the different phases (and tasks within those phases) of the UX engineering lifecycle. In particular, the Requirements Analysis subsection describes and discusses different tasks and techniques for requirements analysis in some depth and from different perspectives. The goal of this chapter is to reinforce the importance of this phase of the lifecycle and to provide real-world examples of the benefits of conducting the kinds of techniques discussed in the later chapters of this subsection.

The following three main topics must be studied and understood to tailor design to support unique UX requirements:

- The users
- The users' tasks
- The users' work environments

In the UX engineering lifecycle, the first is addressed in the task called the user profile, and the remaining two are addressed in the task called contextual task analysis.

41.2.1 User Profile

There is no single best UX style or approach for all types of users. Specific user interface design alternatives that optimize the performance of some types of users may actually degrade the performance of other types of users. For example, an infrequent, casual user needs an easy-to-learn and easy-to-remember interface, but a high-frequency expert user needs an efficient, powerful, and flexible interface. These are not necessarily the same thing. Outward facing websites, unlike desktop software, share the need for an easy-to-learn interface. Even in the cases where websites have expert and frequent users, they must be accessible to new users for the site to grow. On the other hand, intranets may benefit from an interface tailored to a frequent user.

Websites face unique challenges if their success is dependent on objectives such as sales, lead generation, or page views. Usability may not even become relevant to a user if the website fails to engage them immediately by creating the right first impression, causing them to exit within seconds of arriving on the home page. A clear understanding of the intended audience helps designers ensure a website's ability to present its potential value to the user within 10 seconds of home page launch. Many user populations also have specific special needs that affect their ability to interact with software applications and websites. This section (Section A-Requirements Specification, in Part VI-The Development Process) covers the special characteristics of the elderly (see Chapter 35), children (see Chapter 36), cross-cultural audiences (see Chapter 39), and users with various kinds of physical and perceptual disabilities (see Chapters 38 and 40).

The more designers know about the specific characteristics of a population of users (e.g., expected frequency of use, typing skill, learning styles, potential anxiety regarding the defined tasks, etc.), the more likely they will be to make optimal UX design decisions. The purpose of a user profile is thus to establish the general requirements of a category of users in terms of overall UX style and approach.

Adlin and Pruitt's excellent pioneering work on the use of "personas" has become a key tool in building effective user profiles (Pruitt and Adlin 2006; see also Chapter 46). Personas are realistic and detailed descriptions of imaginary people who represent the key characteristics, skill sets, goals, top tasks, responsibilities, tools, pain points, and so on of a category of users. Personas can be used to drive discussion and design throughout the development lifecycle, from conception to user acceptance testing. They help keep the focus on users' needs, preferences, and goals, and they give stakeholders a common language with which to consider, discuss, and evaluate design ideas.

Niche-focused sites may make use of deeper user profile research by using tools such as Myer-Briggs personality profiling studies (Quenk 2009) as an alternative or adjunct to persona development. These tools can help designers focus on how to better motivate or communicate with the target audience.

For example, a site focused on a strongly visual audience, such as professional photographers, would be better served by presenting key information using graphics rather than using primarily text. Similarly, software developer personality profiling tells us to keep messages short and direct, and to avoid "marketese" language or an overly "slick" graphical look. When software developers are led to conclusions via inductive reasoning, they are more likely to remain engaged. Similarly, credibility is established with software developers by credibly establishing a product's acceptance by recognized experts rather than by authority figures whose reputation is not based on knowledge.

Besides understanding individual users of different categories, understanding the characteristics of groups and communities is another aspect of user profiling that becomes important in the design of groupware and products aimed at online communities and computer-supported cooperative work. This aspect of requirements analysis is addressed by Olson and Olson (Chapter 24) in this volume.

The user profile task fits into the overall UX engineering lifecycle as follows:

- The user profile task is the first task in the UX engineering lifecycle.
- The user profile task will feed directly into the contextual task analysis by identifying categories of users whose tasks and work environment must be studied in that later task.
- The user profile task will feed directly into the UX goal-setting task in which UX goals are in part driven directly by user characteristics (e.g., a low frequency of use suggests a need for ease of learning and remembering). Thus, different UX goals will be extracted from the profiles of different categories of users.
- Ultimately, the user profile task will have a direct impact on all design tasks, which are focused on realizing UX and business goals, which are in turn based in part on user profiles.
- The user profile task will also drive the selection of UX evaluation issues and test users.
- Output from the user profile task will be documented in the style guide along with other requirements analysis outputs.

41.2.2 Contextual Task Analysis

The purpose of contextual task analysis is to obtain a user-centered model of work as it is currently performed (i.e., to understand how users currently think about, talk about, and do their work in their current environment). In the case of e-Commerce websites, effective "persuasive" design requires an understanding of what is required to establish vendor credibility and adequate product detail. Without contextual task analysis, designers are likely to overlook important product or vendor information that may be a key to motivate users to make buy decisions.

In e-Commerce website development, designers often fail to address the scope of information a visitor needs to make a buy or engagement decision as well as the level of risk a visitor may face. For example, the risks associated with choosing a family vacation, health insurance plan, or a college education vary greatly. In addition, in B-to-B websites, up to as many as 20 different people are involved in the buy decision (McIntosh 2011), so the role of information sharing in consensus building must be addressed. Beyond the actual end users themselves, the following questions needs to be addressed: what are the roles of the decision makers, what risks do they face, and what information do they require?

Finally, without a truly user-centered approach to information architecture, applications and websites often fail to organize and present data within the UX in a way that satisfies the users' needs for timely information. Thus, the purpose of this task is to supplement more traditional types of system analyses to define UX requirements and to point toward ways to meet those requirements. Then, in later tasks, these requirements can be applied directly in making UX design decisions.

Contextual task analysis consists of the following basic steps:

- Gathering background information to define task scenarios
- Collecting and analyzing data from observations of and interviews with users
- Constructing and validating models of how users currently conduct tasks, which sometimes include making buy or engagement decisions

A central, key step in contextual task analysis is the second step, sometimes referred to as "contextual observations/interviews." Here, the idea is that analysts must observe and interview users in their real-life work context to understand their needs and "hot button" motivators as well as the role of consensus in their decisions.

To revisit the software developer example, task analysis for a website selling training for developers might reveal that the typical they may be motivated to purchase and easily able to move to the buy decision, but that they are notoriously poor "sales" people and their blunt communication style will inhibit their ability to gain consensus from the decision makers who hold the purse strings. Analysis could reveal the typical roles of the other decision makers such as IT management personnel. Only by understanding these roles within the context of the software developer's need to convince them can designers integrate the informational tools necessary to empower the software developer to gain necessary approval and complete the buying process.

Thus, data gathered during contextual task analysis allows designers to effectively structure and present functionality and information in a website or application user interface in a way that taps into users' needs and supports their task objectives.

An abstract modeling of users' tasks, which is the focus of more traditional types of business and systems analysis, runs the risk of making fatally flawed assumptions about key aspects of actual workflow and work environment. That is, traditional systems analysis models work in the abstract, without considering the basic capabilities and constraints of human information processing, the particular characteristics of the intended user population, the unique characteristics of the work environment, and how users themselves model, carry out, and talk about their tasks. The result of this analysis approach is often systems that provide all necessary functionality, but in an organization and presentation that simply does not support the natural flow of work and address the current "points of pain" for individual users.

One generic aspect of tasks that has become increasingly important with the explosion of e-Commerce is the issue of security and privacy. Understanding users' requirements for and expectations of privacy is necessary to make users feel confident about providing personal information in the course of online transactions. Contextual task analyses help provides the necessary insights to enable the optimal design of websites to support the goal of gathering personal data. This important aspect of task analysis is addressed by Karat, Karat, and Brodie (Chapter 29) in this volume.

Contextual task analysis fits in the overall UX engineering lifecycle as follows:

- The user profile task will feed directly into contextual task analysis by identifying categories of users (e.g., business to business purchasers, soccer moms, software developers) whose tasks must be studied.
- Contextual task analysis will feed directly into the UX goal setting task by helping to identify different primary goals for different task types (use cases) and by identifying conversion barriers and weaknesses in the current tools that can be reduced through good UX design.
- Contextual task analysis will feed directly into the information architecture task. Current user knowledge and experience are exploited as much as possible to facilitate ease of learning, reduce anxiety, and ensure that adequate product and vendor information is available to make a buy decision.
- Contextual task analysis will be documented in the product style guide.
- Ultimately, contextual task analysis will have a direct impact on all design tasks, and on the selection of UX testing and evaluation issues, as well as on the design of UX testing materials.

41.3 MOTIVATING AND JUSTIFYING UX REQUIREMENTS ANALYSIS

While Sections 41.1 and 41.2 set the stage for Requirements Analysis within the broader perspective of the whole UX engineering lifecycle, this section provides motivation and justification for investing in UX Requirements Analysis tasks and activities, in particular through the reporting of "war stories" from the experience of the authors and other practitioners. The war stories are divided into those relating to user, environment, and task requirements.

41.3.1 User Requirements

Following are three examples from our own experience of the importance of knowing your users.

Company sites that focus on a narrowly defined target audience may benefit greatly from in-depth persona research. We were approached by a client selling advanced in-house training for software programmers. We suspected that the target audience (software developers) fell into a narrow personality profile and a specific learning style. If this is true then we could identify optimal message styles and key motivators. Our research indicated that developers prefer a straightforward, factual, bulleted, "nonmarketese" message style. We also discovered that as intuitive learners, they prefer to be challenged to deduce conclusions rather than being lectured. For example, we can teach programmers far more by asking them to solve a puzzle or coding dilemma and leading them to the result than if we simply explain a process or technique. In addition, the software developers respect knowledge more than authority, money, or status. During the design of a website to market training to developers, we applied this understanding by establishing the expertise of our trainers clearly and immediately. Text content was delivered in a blunt-bulleted style.

User research also revealed that software developers as a group are not comfortable selling, especially to nontechnical people. Selling is a skill that they disdain. Our courses were expensive and time consuming and in the post dot com recession, we knew that training budgets were being scrutinized. The days of easy management approval for training were past. We felt we had to give programmers the tools to sell to their bosses and convince them that our training had value. We prepared easy-to-forward PowerPoint demos, e-mails they could copy, and cost-benefit reviews. Finally, we prepared a developer-focused guide on "how to convince management to pay for your training." It was a simple, humorous set of suggestions about how to get consensus without becoming a "sales weasel." We knew that the tone had to avoid sounding condescending and it had to be simple and direct. We had to empower them to do something out of their comfort zone and give them the confidence to succeed.

User requirements research also made it clear that we would lose great credibility if the website was developed in any technology other than the one on which training was being offered, which was .net. In fact, a great implementation of the .net technology in the training site was one of the most successful ways to demonstrate that not only did the company offer .net training but they were .net expert users as well.

The results of the website design driven by the user requirements research described above were impressive. When the new website was launched, sales increased immediately, and the word spread that our client was a company that "got it." After only two years in business, the company won a prestigious national Reader's Choice award for "Best Training Company." Their entire marketing campaign had been a nontraditional "word of mouth" campaign based on research done in the user profiling task. The profiling research on developer "learning styles" revealed the pitfalls to avoid and the approach to take. Had we skipped this research, it is unlikely that we would have achieved the viral success that our client company found.

For two other clients, we first interviewed project team members (developers) to get a general sense of the user population. Our purpose was to solicit input to the design of a user profile questionnaire that we would later employ to solicit profile information directly from the target users themselves.

In one case, the project team was convinced that their users would have a generally low level of familiarity with the Microsoft Windows platform they planned for their product. They were thus prepared to depart significantly from the Windows platform user interface standards in their application user interface. The user profile questionnaire, however, revealed a generally high level of Windows experience. This, and the fact that the Windows user interface standard was a good fit for the application functionality, led us to strongly advise the client to adopt the Windows standards as closely as possible. They were still interested in creating their own unique user interface, but early testing of several alternative designs that varied in how faithfully they followed Windows standards clearly showed that users learned much more quickly, the more consistent the design was with Windows standards.

On the other project, the development team felt quite confident that users would generally have high levels of computer literacy. However, an extensive user profile questionnaire of more than 800 users revealed that only a very small percentage of potential users had any experience with computer software at all, let alone the Windows user interface standards. In this case, based on this user profile data, we designed (and validated through testing) a highly simplified user interface that departed significantly from the Windows standards.

In all these cases, the following two things are clear:

1. Project team members often have serious misconceptions about the key characteristics of their target users.
2. These misconceptions could lead teams to design inappropriate UX for those users.

41.3.2 Work Environment Requirements

It is also important to understand the environment—both physical and social—in which users will be utilizing an application or website to carry out their tasks, because this environment will place constraints on how they work and how well they work.

By analogy, suppose a screwdriver is being designed and all that is known is the size of the screw head it must fit. So, something like a traditional screwdriver is designed, with the correctly sized blade. But, it then turns out that the user needs to apply the screw from the inside of a narrow pipe to assemble some piece of equipment. Clearly, a traditional screwdriver will be useless in this work context.

Similarly, suppose a software application is being designed for a set of users, but the designers have never gone to the users' actual work environment. They assume a traditional office-like environment and design software that will work on a traditional workstation. But, it then turns out that in the actual work environment, users are constantly in motion, moving all around the environment to get different parts of an overall job done. If software for a traditional workstation is designed, this will simply not work in this environment. Software that will run on a smaller and more portable device that can be carried around with the user, such as the units carried by UPS delivery staff, would be required instead.

In another example, suppose designers have never visited the user's workplace, and they assume users all work in closed offices. So, a system with voice input and output is designed. But, it then turns out that users work in one big open area with desks located right next to one another. The noise from all those talking people and workstations will create an impossible work environment, and most voice-recognition systems simply do not work with acceptable accuracy in a noisy environment. The point is, there are many aspects of the actual work environment that will determine how well a tool will work in that environment, and so the environment itself must be studied and the tool tailored to it.

A real example of the importance of understanding the users' work environment comes from a project we worked on with a large metropolitan police department. Requirements analysis activities revealed that in the typical police station, the appearance of the interior is dark, run-down, and cluttered; the lighting is harsh and artificial; and the air is close and sometimes very hot. The noise level can be high, the work areas are cramped and cluttered, and the overall atmosphere is tense and high pressured at best, chaotic, and sometimes riotous at worst. These conditions most likely have a general impact on morale, and certainly will have an impact on cognitive functioning that, in turn, will impact productivity and effectiveness. A software user interface must be carefully designed to support the natural and possibly extreme degradations of human performance under these conditions.

In addition, it was observed that in the noisy, stressful, and distracting work environment in a typical police station, users will frequently be interrupted while performing tasks, sometimes by other competing tasks, sometimes by unexpected events, unpredictable prisoners, and so forth. A user interface in such an environment must constantly maintain enough context information on the screen, so that when users' attention is temporarily but frequently drawn away from their tasks, they can quickly get reoriented and continue their tasks without errors, and do not have to backup or repeat any work.

In an example of the importance of the social environment, in redesigning a B-to-B site, our user research revealed that their product typically required consensus from a number of management roles. The initial persuasive focus on benefits and features had been targeted to the primary user and in the process our client company had neglected addressing the information needs and associated risks of the other stakeholders in the decision process. While the target user may desire a product because it will make their work lives much easier, failing to address the total cost of ownership, or the impact of the purchase/engagement on other team members could result in other stakeholders derailing the purchase. We recommended providing the user with information that might be nonessential to him but essential to the other key decision makers. By preparing primary users with the necessary information, we enabled them to answer product questions from all stakeholders.

41.3.3 Task Requirements

Besides the users themselves and the environments they work in, the tasks users do have their own inherent requirements.

On e-Commerce websites, users require complete sets of product and vendor information to be comfortable making a decision to buy. By carefully defining the typical user tasks or scenarios and conducting requirements analyses on those tasks, we are able to identify the range of information necessary to motivate buy decisions. The information requirements can vary greatly from one site to another and even though the tasks may be similar at a high level (e.g., purchase product X or search for product Y and add to wish list) unless a project-specific requirements analysis is conducted prior to design, an e-Commerce website may not be optimized to motivate buy decisions in its unique target audience.

Requirements for similar products can vary for many reasons. For example, for the website of a well-known company with a trusted brand, the need for vendor information may be minimal, and might be as simple as assuring the client that the online sale is secure and privacy will be respected. For companies without a known and trusted brand on the other hand, websites will need to establish vendor credibility through, for example, company history and details, testimonials from other customers, and so on.

When redesigning a website for a leading tour operator, we conducted careful research to identify and prioritize what the users felt the critical data was that they needed to know about the vendor. We began direct user testing on competitive sites and discovered that the first impressions on other sites led users to believe that they were targeting “white seniors.” Minorities felt the sites to be “unfriendly” and exclusive. It was easy to find what elements the vendor information set needed to include to ensure that website appealed to the wider target audience. To build the vendor information set of critical data, we first brainstormed a list of every possible question that might arise about the vendor. We checked that list against questions coming in to the call center sales staff and via e-mail to customer service. Next, we prioritized the information and then identified where on the site this information needed to be available. When the site was relaunched, the customer base increased and both custom and group sales improved. An additional benefit was that questions to the call center reduced dramatically and the percentage of online orders increased. Savings to the call center alone were several hundred thousand dollars in the first year. By detailing the task scenarios and carefully building the persuasive requirements based on direct user research, significant revenue increases were realized. Without this research, opportunities would have either been missed or identified later in the process requiring costly changes.

Clearly, it is possible, even likely, to design a UX that does not support users, their tasks, or their work environments, if an adequate requirements analysis is not conducted prior to design and incorporated into design. The chapters that follow describe and discuss a variety of techniques for gathering these all-important requirements.

REFERENCES

Beaird, J. 2010. *The Principles of Beautiful Web Design*. Canada: SitePoint, Pty., Ltd.

Eisenberg, E., and J. Eisenberg. 2005. *Call to Action: Secret Formulas to Improve Online Results*. Austin, TX: Wizard Academy Press.

Goldstein, N. J., S. J. Martin, and R. B. Cialdini. 2008. *Yes! 50 Scientifically Proven Ways to be Persuasive*. New York: Free Press.

Mayhew, D. J. 1999. *The Usability Engineering Lifecycle*. San Francisco, CA: Morgan Kaufmann Publishers.

McIntosh, M. 2011. Locate Several Decision-Makers Within Each Target Business to Improve B2B Relationships. http://www.buzzle.com/articles/locate-several-decision-makers-within-each-target-business-to-improve-b2b-relationships.html (accessed January 18, 2011).

Nielsen, J., and H. Loranger. 2006. *Prioritizing Web Usability*. Berkeley, CA: New Riders Press.

Pruitt, J., and T. Adlin. 2006. *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*. San Francisco, CA: Morgan Kaufmann Publishers/Elsevier.

Quenk, N. L. 2009. *Essentials of Myers-Briggs Type Indicator Assessment*. Hoboken, NJ: John Wiley & Sons.

Tidwell, J. 2010. *Designing Interfaces*. Sebastopol, CA: O’Reilly Media.

42 Task Analysis

Catherine Courage, Jhilmil Jain, Janice (Ginny) Redish, and Dennis Wixon

CONTENTS

Successful design comes from a marriage of users' goals and (usually) new technologies. Successful design does not necessarily perpetuate users' current ways of working, but it is built on a deep understanding of those ways and of how a new design will change them.

In this chapter, we explore modern interpretations and uses of task analysis.

Section 42.1 defines the meaning of task analysis in this chapter.

Section 42.3 discusses the following practical issues to consider for task analysis:

- Planning for a task analysis
- Collecting task analysis data
- Analyzing and presenting the data

Lastly, Section 42.4 presents real-world case studies of new methodologies and of task analysis applied to a variety of domains.

42.1 DEFINING TASK ANALYSIS

"Task analysis" has different meanings to different authors. Rather than delve into all of the different interpretations, we give a single definition here. For a thorough comparison of task analysis definitions, see Redish and Wixon (2003).

42.1.1 Task Analysis in This Chapter

Task analysis means understanding users' work. Thus, task analysis encompasses all sorts of techniques, including naturalistic observations and interviews, shadowing users or doing "day in the life of" studies, conducting ethnographic interviews, and observing and listening to users who are performing specific tasks. It includes gathering information that leads to insights about users' lives at work or at home, to scenarios and use cases, and sometimes to detailed flowcharts of work processes or specific procedures.

In our view, a major emphasis of task analysis is predesign, and three types of analysis—user, task, environmental—are necessary input to designing any product. Task analysis is, therefore, an integral part of a triangle that covers users, tasks, and environments. As described in more detail in Section 42.2, task analysis goes hand in hand with understanding users (user analysis) and understanding the users' physical, technological, cultural, social, and political environments (environmental analysis).

Users are absolutely critical to all three types of analysis. In our view, *task analysis requires watching, listening to, and talking with users*. Other people, such as managers and supervisors, and other information sources, such as print or online documentation, are useful only secondarily for a task analysis. Relying on them may lead to a false understanding.

Like Kirwan and Ainsworth (1992), we also believe that task analysis does not stop with design. Task analysis continues to be critical at every stage of the design and development process. Task analysis is the major input to use cases and design specifications. Task analysis helps us understand how the emerging product affects users. It is the key to evaluating designs as scenarios for heuristic evaluations and for usability testing. Task analysis must be the organizing principle for documentation and training.

We recognize that efficiency-oriented, detailed task analyses, such as TAG (Task Analysis Grammar) and GOMS (goals, operators, methods, and selection), have a place in evaluating some products, especially those for which efficiency on the order of seconds saved is important (see Gray, John, and Atwood [1993] for example). However, that type of task analysis is not the focus of this chapter. The focus here is a broad understanding of the world in which the new product will be used.

42.2 CONSIDERING FOUR PRINCIPLES THAT UNDERLIE OUR VIEW OF TASK ANALYSIS

The practical advice for doing task analysis later in this chapter is based on the four principles described in this section.

42.2.1 Task Analysis Is an Integral Part of a Broader Analysis

The first principle is that task analysis by itself is not enough to give you the understanding that you need to design or evaluate a product. The methodology you need brings together information about three interwoven elements:

1. Users: Who are they? What characteristics are relevant to what you are designing? What do they know about the technology? What do they know about the domain? How motivated are they? What mental models do they have of the activities your product covers?
2. Tasks.

3. Users' environments: Physical situation in which the tasks occur; technology available to the users; social, cultural, language considerations.

42.2.2 Task Analysis Includes Understanding Users' Goals

The second principle is that a task is what someone does to achieve a goal. Task analysis is concerned with all the stages of the action cycle, as described by Donald Norman (1988):

1. Forming the goal
2. Forming the intention
3. Specifying an action
4. Executing the action
5. Perceiving the state of the world
6. Interpreting the state of the world
7. Evaluating the outcome

42.2.3 Task Analysis Is Relevant at All Stages of the Process

Our third principle is that task analysis belongs everywhere in the process of planning, designing, developing, and evaluating a product. Task analysis, like so much else in the user-centered design process, should be done iteratively. The focus, methods, granularity, and presentation may change over time as different questions and different types and levels of information become more or less relevant. (See Redish and Wixon [2003] for a detailed discussion of task analysis at different stages.)

42.2.4 Practical Reality Impinges on What We Actually Do

Our fourth principle is that in the fast-paced world of software and web design, in reality, what we can do for a task analysis (or any other aspect of user-centered design) depends on many factors. These factors include time, resources, people, availability of users to observe and talk to, and travel restrictions.

This chapter is meant to help you decide the best approaches to consider for whatever situation you are in. We discuss logistics and provide a practical guide on topics such as planning, conducting, analyzing, and presenting data. For a deeper discussion of how to convince clients or others in your company of the importance of usability techniques like task analysis (i.e., selling task analysis), refer to Chapter 59 in this handbook.

42.3 DOING TASK ANALYSIS

42.3.1 Planning for a Task Analysis

42.3.1.1 Background Research

If there is an existing product, learn as much as you can about that product before you begin your task analysis. Having a level of familiarity with the product provides the advantage of running user observation sessions with fewer interruptions.

Information from server logs can help determine what you want to observe in your task analysis. From these records you can learn about visitors, click paths, time per page, exit point, actions completed.

Other sources for information include the product team, marketing, customer support, early adopter feedback, and competitor product. (See Courage and Baxter [2004], Chapter 2, for a detailed discussion.)

42.3.1.2 Getting into the Project Plan and Sign-Offs

A critical aspect of being able to do task analysis throughout a project is getting this and all other usability activities into the project plan. The extent to which this can happen depends, of course, on buy-in from the project team, but also on other factors, such as whether project plans exist and how much detail is specified. The more strongly a project team uses a formal project plan, the more critical it is to get usability activities, such as task analysis, into the plan. Time, resources, and respect from managers and developers for the information may be dependent on being part of the formal plan.

Another approach that many usability specialists follow is to create a usability project plan, which parallels the system design and development project plan. That is fine if the system people understand and respect the parallelism of the plans.

Whether the usability plan is part of the overall project plan or a parallel track, it is important to get sign-off from the rest of the design and development team with respect to activities that the usability team will do, resources needed for those activities, information that will be brought back from those activities, deliverables (formal or informal) that will come from that information, and dates for those deliverables.

42.3.1.3 Providing Useful and Usable Data

As usability specialists, we help project teams only if we provide useful and usable data when they need it. Therefore, as noted, staying closely aligned with a project plan and schedule is critical.

What else, besides timeliness, makes data useful and usable?

- *Data the project team needs.* As usability specialists, we should approach any project looking for where the team needs data about users (whether they know it or not) and plan to collect, interpret, and present that data in a way that the team can understand and directly use. If there is no rationale for how the data will impact the product (now or later), it is best to invest your time elsewhere.
- *Data that is credible.* Time pressures and limited resources often curtail the extent of any usability activity, including task analysis. In general, we follow the maxim that some data about users and their

work is better than no data. And practical experience tells us that is true, as long as the data is accurate and representative. Good data is collected in a systematic, careful, rigorous way from appropriate users. To do this successfully, you must become familiar with the product and domain. Domain familiarity will not only allow you to identify the appropriate type of task analysis and to formulate the most appropriate questions and observations, but the team will also take you more seriously.

42.3.1.4 Deciding What the Project Team Needs

The task analysis that you want to do will depend in part on the type of product or website that you are working on. Consider at least the following six factors as you think about the project for which you are planning task analyses and other usability activities.

42.3.1.4.1 Where Is the Product in Its Overall Lifecycle?

Upgrading an existing product without changing the medium (a new software release; a revision of a website)? Changing business processes or medium (going from a stand-alone application to a web-based or cloud-based one)? Developing something totally new?

42.3.1.4.2 How Much Time Do You Have to Conduct Your Task Analysis?

Has the team always been onboard with task analysis and they defer to you to determine the appropriate activity and time line? Has the team suddenly discovered that they need some data quickly and they have not allotted for it in their schedule? Does the team believe that there is no time in the schedule for task analysis?

42.3.1.4.3 How Broad or Specialized Is the User Population for the Product?

Is the product for a very broad public market, niche business market where you can easily define the user population and access to them is through account executives (or similar), or special audience (children/the elderly/persons with disabilities, etc.)?

42.3.1.4.4 How Widespread Geographically, Culturally, and Linguistically Is the User Population for the Product?

Is the product global (how far/how many countries/cultures/languages)? Or is it local?

42.3.1.4.5 How Detailed Must We Specify the Tasks?

Is the product safety-critical where tasks are very specific and must be done in specified ways? Do users receive training until they prove their competence in completing the tasks accurately and efficiently? Will it be used by many different types of people for different tasks, which they may do in different ways?

42.3.1.4.6 Is This a Special Type of Product for Which Traditional Task Analysis May Not Be Useful?

Some applications do not fit the traditional approach for task analysis. These are primarily applications that the user does for fun, for example, games. Traditional task analysis would not aid much in the design and development of these applications. As a result, game designers do not typically do task analysis. They are more interested in mood, theme, story, drama, progression, surprise, pacing, and the physical correlates of these experiences. (Also see Chapters 31 and 34 in this handbook.)

42.3.1.5 Deciding on an Appropriate Level of Granularity

Another aspect to consider as you plan a task analysis is the types of analysis to do. Understanding users' goals and their work can be done at several different levels. You might be interested in one or more of the following types of analysis:

- *Analysis of a person's typical day or week*: "A day in the life of" or "an evening at home with"
- *Job analysis*: All the goals and tasks that someone does in a specific role—daily, monthly, or over longer periods
- *Workflow analysis*: Process analysis, cross-user analysis, how work moves from person to person
- *High-level task analysis*: The work needed to accomplish a large goal broken down into subgoals and major tasks
- *Procedural analysis*: The specific steps and decisions the user takes to accomplish a task.

42.3.1.6 Deciding Where to Start

You must first understand how far along in the process the project is. Unfortunately, by the time usability specialists know about the project, the strategic planning and predesign stages may already be considered closed. The project may be at considerable risk if the strategic planning and predesign questions were never answered or were answered based on speculation or internal discussions without users. However, it may be unproductive to spend time and effort collecting data that speaks to those questions if no one is willing to listen to the answers you bring back. If that is the case, a more productive use of the limited time and resources for task analysis and other usability activities would be to understand where the project is and how to influence it from that point forward. In some cases the team realizes (often with your help) that they do need to take a step back to the earlier stages and collect user data in order to prevent the product from going down the wrong path.

42.3.1.7 Gathering Reusable Data

Time, resources, and costs are likely to limit the number of times you can return to users for task analysis. One good approach is to collect extensive data about users' work in a relatively holistic way, capturing that data on video, audio, or in extensive notes so that you can return to the data—rather than to the users—with different questions in mind at different times.

To do this, an open-ended field study method combined with detailed information gathering is best (Wixon 1995). Also, having a relatively detailed log of the raw data is necessary (videotape, audio tape, or verbatim transcripts).

42.3.1.8 Going to Different Users at Different Times

Even if you gather extensive data holistically on early site visits and return to the data, you may not have the information that you need. In that case, go out again—and go to different users.

You can use each set of site visits not only to answer the specific immediate issues and questions but also to enrich the team's general understanding of users, their work, and their environments. Although your immediate focus may be a specific why or what or how question, always drill down so that you are, in fact, seeing the why behind the what or the what and how behind the why.

42.3.1.9 Preparing for International Task Analysis

Selling an international study to management can be hard. Refer to Courage, Redish, and Wixon 2007 for practical issues to consider. International studies add several significant considerations including translators, recruiting, logistics, and background research.

42.3.1.9.1 Translators

If the language of your study participants is not your native language, you will want to employ a translator. This person will need to translate all of your tasks analysis materials beforehand and then translate in real time during the session (simultaneous translation). Simultaneous translators are often known as interpreters, and interpreting is a specialized and expensive skill, as it requires a great deal of concentration and focus over extended periods of time. You will also need to allow time to train the translator in the goals, objectives, and procedures of your study to ensure that as little as possible gets lost in translation.

42.3.1.9.2 Recruiting

Recruiting for an international study is not the same as recruiting for a local study. Factors that often differ include appropriate ways to contact users, recruitment and incentive costs, no-show rates, the times that people are willing to participate, and holidays. Be sure to account for these differences in your plan. If possible, it is wise to engage a local recruiting company. If your company is international, engage the people in the office of the country you will be visiting to help you find users and to learn about the culture in advance (Siegel and Dray 2005; Dray and Siegel 2005).

42.3.1.9.3 Logistics

If you are visiting multiple locations, time your visits so that you can minimize your travel time and costs. You do not want to fly roundtrip to Hong Kong from New York and then 2 weeks later fly to Shanghai from New York. Also, be sure to give yourself some time to adjust to new time zones. Task analysis activities require you to be very alert and attentive. You will not be of much value if you are completely exhausted.

Also consider your equipment. Will it work in the country you are traveling to? Will it be too much to travel with? If you need to scale back, consider this early on. Should you rent the necessary equipment? If yes, where? Thinking through all of the details ahead of time will save headaches as your site visit approaches and while you are on site.

42.3.1.9.4 Background Research

Do as much background research as you can before you go. This will help prepare you for some of the things you may encounter. For example, read travel guides and books on cultural etiquette. Read articles in research papers, professional magazines, and on the web. Examine competitor products that may be thriving in the culture you are going to visit. Use this information to guide you as you create your task analysis plan.

If this sounds like there is a lot of preparation to conduct international task analysis, that is because there is. The good news is that the preparation is worthwhile. As you come to understand a culture and make connections with people from that location, future studies will be easier and less costly. And the understanding that you get from an international task analysis will pay off richly in the product's success in that culture.

42.3.2 Collecting Task Analysis Data

While traditional task analysis focused on time and motion studies, modern task analysis relies more on ethnography and cognitive psychology; and that is the focus we take in this chapter. (For a more detailed discussion of task analysis roots in ethnography and cognitive psychology, see Redish and Wixon [2003].)

For the basics on how to select users and environments, making a convenience sample as representative as possible, conducting site visits, and observing and interviewing users, refer to Courage, Redish, and Wixon (2007).

42.3.2.1 Rapid Task Analysis

As we have said, sometimes, the team needed the data 2 weeks ago; and sometimes, they just do not have the time or resources for a multi-week task analysis. Several rapid data collection methods have emerged for these situations. We discuss three here: narrowing the scope, using multiple teams, and scheduling your sessions around activity peaks.

Note that rapid task analysis is not right for all situations. If your team is designing or redesigning a very complex system, there may be so much to understand that even a condensed method will be time consuming. (See the Section 42.3.3 for a discussion of rapid data-presentation methods.)

42.3.2.1.1 Narrow the Scope

One way to do rapid task analysis is to reduce the scope of your study. Instead of trying to understand everything about the users and all of their tasks, focus on collecting data that you will be able to use immediately to make decisions, especially when you have specific issues or questions that need to be resolved.

42.3.2.1.2 Using Multiple Teams

Another way to do rapid task analysis is to have multiple teams collect the data to shorten the time required. This is a great way to engage the entire project team. It also may be necessary in situations where your study is international and involves different languages and cultures. Involving multiple data collectors is ideal if you are working with a team that understands task analysis.

If you take the team data collection approach, you need to make sure that everyone is on the same page about what will be observed and the types of data that will be collected. Templates are ideal in this situation. We recommend the following templates:

- A protocol that gives the procedures and steps that everyone is to follow when with the users. A useful protocol includes information on how to introduce the session to the participant, how long each part of the session should last, how to end the session, and so on.
- A list of questions to ask either during or after the session.
- An observation guide that includes the issues observers should focus on. This helps guide the observations to ensure that everyone is observing the same types of activities. It is different from the list of questions to ask the user.
- A template for recording data. This helps cue observers as to what behaviors they should be observing and also provides a consistent data collection method. It makes analyzing data much easier.
- A template for reporting on the session. This is a standard worksheet that each observer completes at the end of each session to highlight some of the key observations or discoveries made. It can be very useful when the team first regroups to discuss the data. In some cases, teams create templates based on the form of analysis required by the development process.

If the data is being collected over a number of days, check in with your team at the end of each day to talk about key observations. This can be in-person, by phone, or through web-based voice-over-Internet programs. It is great to discuss when the data is fresh. A daily check-in also gives you an opportunity to resolve any procedural issues that observers may be having.

42.3.2.1.3 Schedule Your Sessions around Activity Peaks

If you schedule your visits during times of peak activity, you may observe more activity during a shorter window of time. While this technique is useful in any task analysis, it may be particularly helpful if you are doing rapid task analysis.

If you plan to go during a time of peak activity, you may want to pay particular attention to our general recommendation that an observation team include two people. In a time of peak activity, one person may not be able to keep up with the potentially rapid pace. In our two-person teams, one observer focuses on asking questions, while the other takes notes.

Be aware, however, that many users may not want you to come at times of peak activity precisely because they are so busy and stressed. They may feel that they cannot cope with any distractions or additions to the environment. If seeing times of peak activity is crucial to your issues, you may need to convince users that you will observe unobtrusively and not get in the way. You may need to forego some of the questions and interview time that you would like to have.

42.3.2.2 Bringing Users to You (Field Studies in the Lab)

The best task analysis, of course, is done in the users' context. However, time constraints, travel restrictions, or security restrictions may make it impossible for you to go to the users. You can still do task analysis. Have the users bring their tasks to you.

Although you will not get a true environmental analysis (seeing the users work in their own settings), you can still get a lot of data about users and tasks. When you do task analysis in the lab, make sure you ask the users to bring artifacts relevant to the task being discussed (Butler 1996). These are any objects that participants use to complete their tasks or that result from the task. Artifacts could be a calendar, a procedure manual, a physical form, a computer-generated report, and so on.

42.3.2.2.1 Using the Lab for Procedural Task Analysis

For detailed procedural task analysis (getting down the steps and decisions in completing a specific task), the lab may be the right environment. If you want to capture details, you probably want a videotape record. It is easier to videotape in the lab than to cart equipment to users' sites, set it up, and take it down—although we do recommend videotaping for most task analysis site visits.

Procedural task analysis can be done either individually or with a group of users. In an individual task analysis, you work with one user at a time and delve into exactly how that user completes the process. In a group task analysis (GTA), you focus on four to six users together and gather the task flow from the group.

Group task analysis has several advantages, including the following three. (Also see Courage and Baxter [2004], for more on group task analysis.)

1. *Seeing the details.* Describing a task flow can be difficult. It is easy to overlook routine steps or to forget about some of the details. Like focus groups, a GTA generates group discussion and a synergy among the participants. Because the users are comparing how

they each complete the task, details tend to emerge in the discussion.

2. *Coming to agreement*. One of the goals of a group task analysis is for the participants to come to agreement on a single task flow that represents a common way that they can work to complete the task. When conducting individual task analysis sessions, the researcher must combine the data and make compromises, rather than having the users make these decisions.
3. *Saving time*. You can often develop a rich task flow with a group of users in 2 hours where you would spend 8–10 hours with individual users. This makes group task analysis another good technique for rapid task analysis.

42.3.2.2.2 Using the Lab for Quantitative Task Analysis

If you want quantitative measures, such as time to complete a task under ideal conditions, the lab may again be the best place to do the study. Someone might ask, in that case, whether you are doing a field study or a usability test, but in fact the line between the two techniques is quite blurry. Both can be done in the field or in the lab. Both can be done on old products or new prototypes. Both can be done qualitatively or quantitatively. Both can be done on the user's own work or on scenarios given to the user. Task analysis is a major part of both field studies and usability testing.

42.3.2.3 Remote Task Analysis

If your focus is software or the web, you can do remote task analysis to collect useful data. You can use a screen-sharing application to see the user's screen as the user works. As with lab studies, you cannot observe the user's physical environment. However, one advantage you have over a lab study is that the users are still in their own environments, you can ask users to describe their environments; and some may even be willing to send photos, which can give you useful insights.

Remote task analysis is sometimes your only option, especially in international studies. You may not have the resources for global travel. Doing remote task analysis is most successful when you already have some experience or familiarity with the users' culture.

A remote task analysis can also be very useful to supplement on-site sessions. If you only have the resources to visit one or two users, do so; and then supplement your data through remote sessions with other users.

42.3.2.4 International Task Analysis

As you collect data internationally, you should add two critical considerations to your repertoire: being culturally sensitive and being aware of elements that differ in different countries and cultures.

42.3.2.4.1 Being Culturally Sensitive

Research shows that methods based on the Western thought can lead to cultural conflict and misinterpretation of data, particularly in Asian cultures (Chavan 2005; Ann 2004).

For example, Chavan (2005) explains that, in India, users often hesitate to say that something is bad or to identify issues with a product. Even if an Indian user does not highlight any problems, you cannot assume that no problems exist for that user. Also, users in India may be more likely than users elsewhere to feel that they are being evaluated just because someone is observing them.

For another example, Ann (2004) says that, in China, relationships are very important. Friendship is a prerequisite to dealing in business or finance. This may make it difficult to recruit participants unless they come from friends' referrals. If you are not connected via a friend, users may not want you in their homes. Even if you are able to enter a home without such a connection, you may not hear the users' true feelings, as they may take great care not to offend you. If you do not continue the friendship after the study, Chinese participants may feel used and decline to participate in future studies. Furthermore, a Western researcher and participant relationship would not be well received in China. The Western approach in which you ask questions in a very scientific manner and re-ask them in different ways to probe deeper may lead to distrust rather than to useful information. Less intrusive styles may be more appropriate.

Therefore, choosing a culturally sensitive approach to task analysis for the location you will be visiting is critical. This is where local consultants or people in your company at that location can be invaluable. Whether you actually conduct the study or have them do it, they can be extremely valuable in helping you refine your approach so that the experience is pleasant and comfortable for the users and so that you collect valuable and accurate data.

42.3.2.4.2 Being Aware of Elements That Differ in Different Countries and Cultures

As Siegel and Dray (2005) recommend, you should be aware of the many elements that change from country to country and culture to culture so that you can collect data on whichever of these are relevant to your product. The following are just a few of the elements to consider:

- Purchasing dynamics and financial transactions—for example, forms of payment, willingness to disclose financial information to others or online
- Social structure and service expectations—for example, society classes, willingness to "do it yourself"
- Mental models of geography—for example, address formats, maps
- Use of physical space—for example, size of workspaces
- Climate and environmental conditions—for example, how environmental conditions impact the user of the product

42.3.3 Analyzing and Presenting the Data

Once you have collected the data, of course, you must analyze it and present it. Data is of no value if you do not communicate what you have learned to the people who need the information. To make the data useful, you must bring the data together, think about what you have learned, and draw out implications.

42.3.3.1 Analyzing the Data

Consider the following four principles as you plan to analyze the data.

42.3.3.1.1 Involve the Design Team

Making the effort to involve the rest of team pays off handsomely. Involving other team members ensures that they have a stake in the results. It also allows them to work with the raw data, which helps them internalize the "work of the user" more completely, even if they did not get to participate in the site visits. When you involve the team in the analysis, you build a shared understanding of what was seen and heard at the customer or user sites.

It also ensures that the questions the team has get answered. Often teams refine and redirect their thinking as they go through an analysis. They may drill down more deeply into the data. They may completely change the questions they have of the data. They may change their thinking about the direction they had planned for the product.

Working as a team can also help you generate actionable items that result from the study. You can work together to prioritize the findings. This is ideal to do as a team because you can discuss limitations, such as time, and determine what is truly feasible.

42.3.3.1.2 Make It Traceable

Any analysis should include references back to the raw data. There are many advantages. First and foremost, keeping the link ensures the integrity of the analysis. It is important to be able to say to those who were not involved in collecting and analyzing the data that all conclusions are traceable back to statements by users or direct observation of their behavior. Second, as we noted in discussing the first principle, interpretations may change as analysis progresses. During that process, it is important to be able to revisit the data and recall the context of the behavior or the comments. If you set up ways of tracking the data through analysis, the extra time and effort need not be substantial.

42.3.3.1.3 Make It Visible and Accessible

The analysis may be complex and detailed, but a report laden with text will almost certainly go unread. There is not time in the rush of a project for people to read. Therefore, many teams choose to display their analysis on a wall or in a "war room"; where team members can review and add comments to the display. (See, e.g., Simpson [1998].)

An alternative is a hyperlinked document in which higher level conclusions are linked to more specific analyses. Often the analysis is graphical so that designers can stand back and "see" patterns in the data. Some people create multilevel documents with a one-page summary, supported by a three-page overview, which in turn is supported by a 50-page report.

42.3.3.1.4 Match the Form to the Questions, the Stage, and the Team's Needs

The cardinal rule of all documentation is to give users what they need in the form they need it when they need it. That is why most technical communicators have moved from writing extensive tomes that people do not open to helping teams bring communication into the interface.

The same principle applies to the internal working of any project team. The best form in which to represent the data depends on many factors, including the questions that were asked, the stage in the project's lifecycle (that is, how the information will be used), the time in which the information is needed, and the team and company culture.

Keep in mind the essential purpose of any analysis. Analysis provides an anchor from which designs can be generated and against which they can be evaluated. The analysis is not an end in itself. You must keep the design team engaged with the analysis and with the representations of the analysis. In the Section 42.3.3.2, we describe a few of the possible representations.

42.3.3.2 Presenting the Data

The following are 13 ways to present task analysis data. (Also see Hackos and Redish [1998], especially Chapter 11. Courage and Baxter [2004] describe data analysis for a variety of task analysis techniques. Miles and Huberman [1994] also describe the rich variety of ways to organize qualitative data for interpretation.)

42.3.3.2.1 Affinity Diagrams

Affinity diagrams are hierarchical pictures of user data. They are produced inductively by grouping similar data elements together into categories and then grouping the categories together.

Affinity diagrams derive much of their value from the process that produces them, that is, a deep engagement with the data combined with recurring reflections on the generalization that best captures a number of data elements. Also, teams often produce "collateral" elements while creating an affinity, such as design ideas and additional questions that are captured and then used in design or further data gathering.

42.3.3.2.2 Artifacts

Some types of tasks are deeply intertwined with their artifacts. For example, the task of making appointments is necessarily interwoven with calendars that show dates and times. As a result, it is often best to begin with existing artifacts (e.g., Ellen's appointment book) and to organize data around that representation.

42.3.3.2.3 Flow Diagrams

Flow diagrams answer questions about how information or artifacts flow through a system (process analysis). They illustrate the dependency between system elements or states of the system and what needs to be transferred or moved from one part to another. They also show how roles are divided within an organization as data moves from one person or department to another or between the organization and outsiders.

42.3.3.2.4 Personas

Personas are composite archetypes who represent the primary and secondary users of the product. (Pruitt and Adlin 2005; Mulder and Yaar 2007; Courage, Redish, and Wixon 2007; Pruitt and Adlin 2010). A persona description often includes the user's activities, knowledge, and tasks in some depth. Some people, in fact, also include information about the persona's environments (physical, social, cultural, technological), thus capturing all the triangulated data that we discussed at the beginning of the chapter.

Rich persona descriptions that encompass user, task, and environment information are particularly useful for commercial products, which often begin with market segmentation that classifies and describes potential customers. The task analysis builds on this data by characterizing these users more precisely (Lee and Mikkelson 2000).

Because personas instantiate users as actual people, they tend to be memorable. You can ask "How would Julie do her work if we design the product this way?" (See Cooper [1999].) Personas can come to team meetings as life-sized cardboard figures (Butler and Tahir 1996), on posters or on placemats. They can become part of the product team's e-mail group with someone in charge of representing them (see Chapter 46 in this handbook).

42.3.3.2.5 Scenarios

A scenario is a short story of a specific situation that is real and relevant to a user. A scenario gives the team the user's goal and specific needs. It often also gives the team the user's names for objects and attributes of those objects. It may give the team information on what the user values. (e.g., Is price more important than choice in renting a car.)

Each situation that you observe on a site visit is a scenario. You can also collect scenarios by interviewing users through the critical incident technique (Flanagan 1954) in which you ask users to recall a specific incident and then to tell you about it.

You can elaborate a scenario with the sequence diagram (flowchart) of the procedure the user went through to accomplish the scenario. If accomplishing the scenario is difficult, and the scenario is important, creating a more efficient procedure could become a requirement for the new product. (See also using stories from task analysis and, indeed, throughout design and evaluation, Quesenbery and Brooks [2010].)

42.3.3.2.6 Sequence Diagrams

Flow diagrams track work through a system or across people. Sequence diagrams use time to track the actions and decisions that a user takes. Sequence diagrams (procedural analysis) show what users do and when and how they do it. This type and level of information is critical for interface architecture and design because it gives us the functions, objects, and attributes of a system (e.g., menu items and dialog box design) and the navigation for a website.

By laying out the sequence diagram that represents what users do today, you can often see ways to make the product help users be more efficient and effective. Thus, the sequence diagram of the reality of what you find in predesign observations is often elaborate and messy. It may be important for the team to see that reality as they work toward a more useful product.

42.3.3.2.7 Tables

Tables are an excellent way to show comparisons and so are useful for presenting many types of analysis. Technical communicators, for example, have traditionally used a user/task matrix to understand which tasks are done by which types of users. The user/task matrix becomes a major input to a communication plan—to answer the question of what tasks to include in documentation for people in different roles (e.g., system administrators, end users). Tables can be used to show the relationships between any two (or more) classes of data. See Table 42.1 for an example of a table with three classes of data.

42.3.3.2.8 User Needs Tables

Kujala, Kauppinen, and Rekola (2001) developed what they call a "user needs table" as a way to present a current task sequence along with the problems and possibilities that the designers should think about for each step in the sequence. They hypothesized that it would be easier to use findings about users' needs in design if the findings were connected to the task sequence that forms the basis of use cases. In their studies, designers found this type of presentation very useful in moving from data to requirements. They also found that writing use cases from this type of presentation keeps the use case in the user's language and keeps the use case focused on the user's point of view. Figure 42.1 shows an example of a user needs table.

42.3.3.2.9 Mood Boards for Quick Data Dissemination

Mood boards (Foucault 2005) are an ideal way to present preliminary research results when your team is under a tight timeline and you want to get them some data quickly or when you want to keep the momentum from the study going.

Mood boards are a collection of raw data and artifacts presented with the goal of inspiring the product team early in the development cycle. They often take the form of a large poster showing photographs, sketches, screenshots, participant quotes, and artifacts from the field, such as forms, printouts, and post-it notes. You can use mood boards to highlight some of the interesting or unexpected findings from the study.

You can put together a mood board quickly because the date is not yet refined or synthesized. Mood boards can also be ideal for international studies because they can convey differences between cultures and can give designers a feeling for the aesthetics of the other culture. Figure 42.2 is an example of a mood board from a study that was part of a project to develop technology for older adults in the United States. Mood boards were created for areas where there were product possibilities. These included older adults in the city, fitness for older adults, and fraud against older adults.

TABLE 42.1
Matrix for Weighing Market Size, Users' Tasks, and Potential Product Functions

Users	Relative Market Size High—3 Medium—2 Low—1	High-Level Task	Task Relative Importance High—3 Medium—2 Low—1	Tool Bar Customization	Better Drag and Drop	Progress Meters	Interruption Protection Recovery	Total (Sum of Products, How Well Market Is Served by These Features)
Techno Bob	1	Customize my screen	3	2	1	0	0	9
		Download information	3	0	0	1	2	9
Newbie Ed	1	Customize my screen	1	–1	2	0	0	1
		Download information	3	0	2	3	0	12
Practical Sue	3	Customize my screen	2	2	2	1	0	36
		Download information	2	0	2	1	2	30
Sum				17	35	27	18	

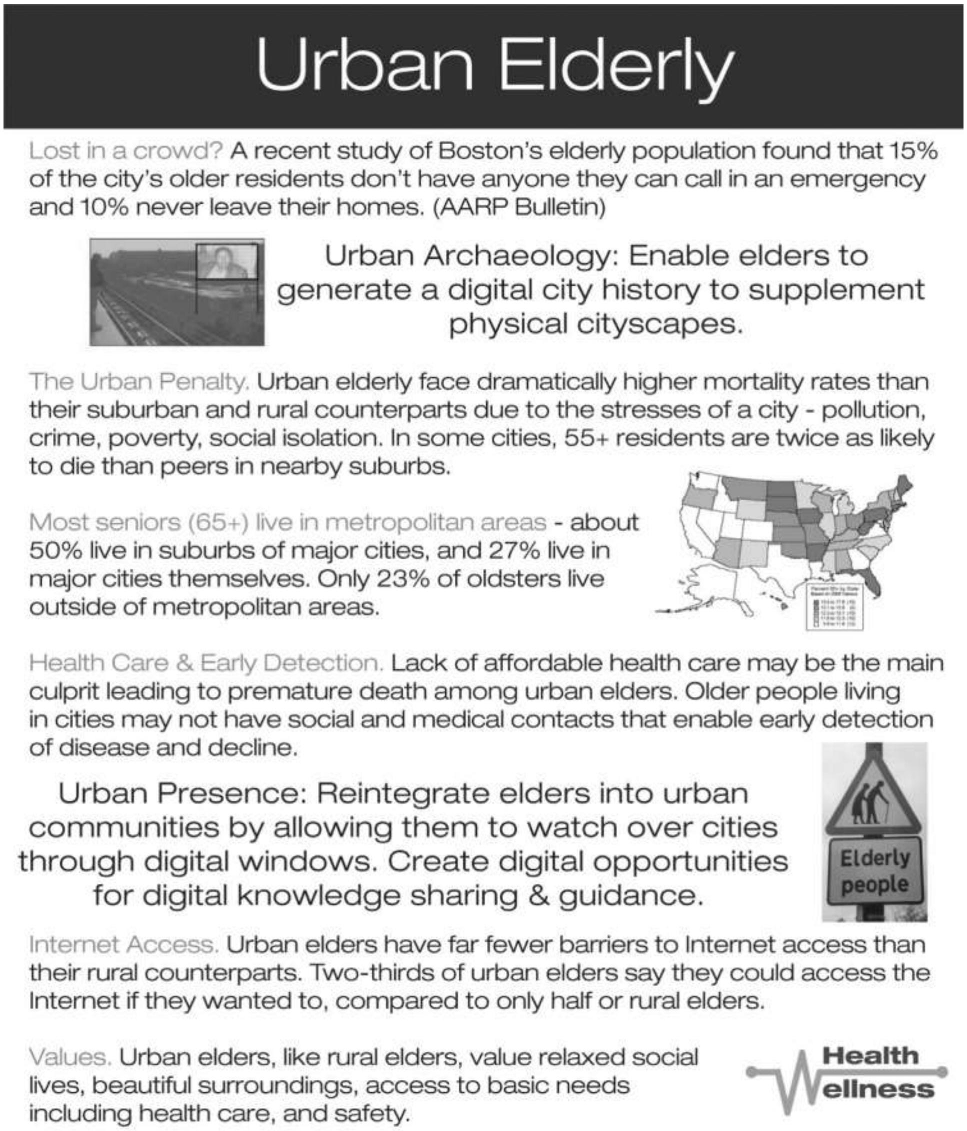
–1—will confuse users
0—users will not use it
1—users mildly positive
2—users strongly positive

Task Sequence	Problems and Possibilities
Step 1: When trapped in an elevator, passenger makes an emergency alarm.	• Passengers want to get out of the elevator as soon as possible. • All kinds of passengers must be able to make an alarm call (blind, foreigners, etc.). • Sometimes passengers may make false alarms unintentionally. • Passengers may be in panic. • Passengers need instant confirmation that they have created a connection to the service center operator and that they are going to get help.
Step 2: Unoccupied service center operator receives the emergency alarm call and asks for information (description of the failure).	• Different versions and types of remote monitoring systems. • Passenger is the only information source. • Service center operator does not notice the emergency alarm call.
Step 3: Service center operator completes transmission of information to the system and sends it to the area serviceman.	• Laborious phase for the service center operator. • Simultaneous calls must be differentiated. • Serviceman cannot see all information. • Inadequate information from a site system. • Possibility: Instructions as to how to operate the system. • Possibility: Possibility to open phone line from call center to the elevator.
Step 4: Service center operator calls the serviceman and reads the description of the failure.	• Extra work for the service center operator.

FIGURE 42.1 An example of a user needs table. (From Kujala, S., M. Kauppinen, and S. Rekola. 2001. Bridging the gap between user needs and user requirements. In *Proceedings of PC-HCI 2001 Conference*, Patras, Greece. With permission.)

42.3.3.2.10 Live from the Field: Another Technique for Quick Data Dissemination

Another very effective rapid communication method is to send information back to the team from the field in real time (Lovejoy and Steele 2005). This method is particularly useful for international studies where the product team is often quite curious and excited about the study. Also, if you are going to be traveling for an extended period, you can get some data to the team so they do not have to wait for weeks to hear what you are learning. Photos, stories, snippets of data, and insights can help the team to feel a part of the study.



Urban Elderly

Lost in a crowd? A recent study of Boston's elderly population found that 15% of the city's older residents don't have anyone they can call in an emergency and 10% never leave their homes. (AARP Bulletin)

Urban Archaeology: Enable elders to generate a digital city history to supplement physical cityscapes.

The Urban Penalty. Urban elderly face dramatically higher mortality rates than their suburban and rural counterparts due to the stresses of a city - pollution, crime, poverty, social isolation. In some cities, 55+ residents are twice as likely to die than peers in nearby suburbs.

Most seniors (65+) live in metropolitan areas - about 50% live in suburbs of major cities, and 27% live in major cities themselves. Only 23% of oldsters live outside of metropolitan areas.

Health Care & Early Detection. Lack of affordable health care may be the main culprit leading to premature death among urban elders. Older people living in cities may not have social and medical contacts that enable early detection of disease and decline.

Urban Presence: Reintegrate elders into urban communities by allowing them to watch over cities through digital windows. Create digital opportunities for digital knowledge sharing & guidance.

Internet Access. Urban elders have far fewer barriers to Internet access than their rural counterparts. Two-thirds of urban elders say they could access the Internet if they wanted to, compared to only half or rural elders.

Values. Urban elders, like rural elders, value relaxed social lives, beautiful surroundings, access to basic needs including health care, and safety.

FIGURE 42.2 An example of a mood board representing urban elderly. (Courtesy of Intel.)

Lovejoy and Steele (2005), for example, collected photos and narrated the photos with data to create what they call photo stories. Other useful ways to communicate findings quickly include blogs, e-mails, and websites.

42.3.3.2.11 Culture Cards

The last three data presentation methods focus on conveying the culture of your study participants. These techniques were developed by Intuit (Foucault 2005) to convey observations from a task analysis study.

Culture cards are a deck of 20–30 physical cards that are used to communicate key findings and images from different cultures. Information such as photographs, brief research findings, user descriptions, or users' needs are printed on materials of cultural significance, such as greeting cards, textbook paper, or handmade paper from the area. Figures 42.3 and 42.4 are examples of culture cards created for a project on pregnancy and parenting.

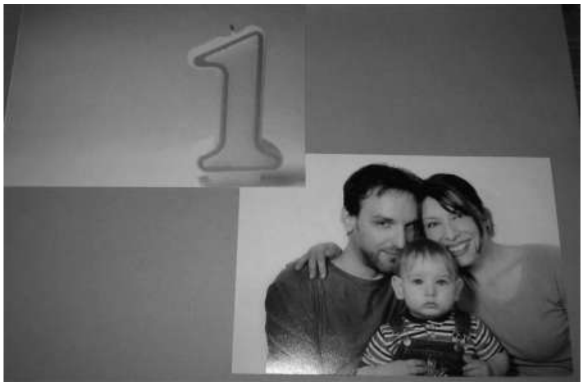


FIGURE 42.3 Images displayed on the front of the culture cards.

The intent is to be motivational: to inspire the product team to create culturally sensitive and appropriate design solutions. Culture cards can provide a useful overview of key findings and serve as a launch pad for brainstorming

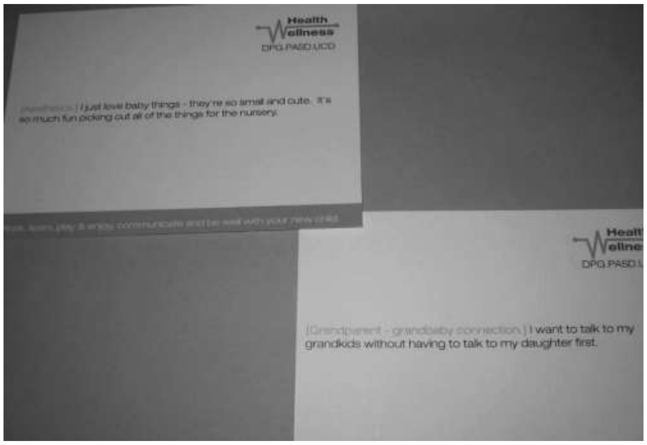

Health
Wellness
DPG.PASD.UCD
I just love baby things - they're so small and cute. It's so much fun picking out all of the things for the nursery.
communicate and be well with your new child
Healt
Wellne
DPG.PASD.U
[Grandparent - grandbaby connection] I want to talk to my grandkids without having to talk to my daughter first.

FIGURE 42.4 Findings displayed on the back of the culture cards.

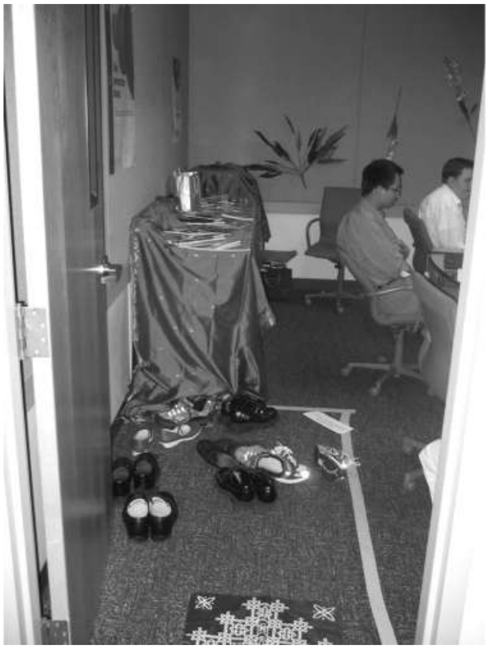


FIGURE 42.5 Photo of a culture scape. Trying to immerse the user in the Asian culture by having the team remove their shoes and by decorating the conference room with artifacts. (Courtesy of Intel.)

and design sessions. Two key advantages are that they break the data into digestible chunks and they cue the designer to appropriate aesthetics for each culture.

42.3.3.2.12 Culture Scapes

When communicating results from an observational study, it is challenging to convey the true feelings and sensations generated by the environment. Smells, sounds, textures, and flavors are all important aspects of the users' environment, but they can be difficult to relay to the team (Foucault 2005). Images and words are often not enough. Figure 42.5 is an example of a culture scape.

Culture scapes are designed to allow the team to richly experience the culture by engaging multiple senses. Foucault (2005) successfully used culture scapes when relaying findings from a mobile technology study in Singapore. During the study the observers were struck by the intense heat and humidity and how it related to their observations. To convey these environmental realities, they presented their findings in a room that was 90 degrees Fahrenheit with 85% humidity. As a result of the simulated environment, team members "got" the importance of these environmental factors and their critical impact on design. If you plan to use culture scapes, it is important while you are in the field to consider what sensory experiences are most important and how you will convey them when you return.

42.3.3.2.13 Culture Capsules

Culture capsules showcase unfamiliar elements of a culture by physically re-creating a space from the observed environment (Foucault, Russell, and Bell 2004). This presentation method is of particular importance when space is the primary consideration in design. Photographs taken in the field guide these re-creations, which may also include artifacts from the physical environment being re-created. For example, Foucault, Russell, and Bell (2004) used this technique to highlight the space constraints in a typical Chinese home so the team would understand the limited space available in a typical home office in China. By replicating the physical spaces they had seen, the researchers made the necessary information about space constraints obvious to the product team. The space also served well as a brainstorming area and a project showcase.

DAN SZUC'S INSIGHTS ON TASK ANALYSIS AND USING TASK ANALYSIS

"Let's All Take a Step Back ... for Just a Moment Please"

Over many years I have been fortunate to be able to be exposed to different projects, roles, people, domains, products, teams and user research approaches. A key component of all this work is the ability to not just do the research but to take the learnings and do something with them. We have also learned the need to make the results come alive and move them out of reports and PowerPoint decks into something real and tangible for product teams.

You Don't Work in a Research Bubble

One of the challenging parts of analysis from field studies, interviews, testing, and so on is taking everything you have learned and bridge it into the design. It's nice on projects to be given the time to do some up-front user research, to have the space needed to analyze the data, both mentally and in physical space, to look for insights and think about how this can be communicated through design. Often we get lost in the methods themselves and we forget the journey we are on, the goals of the research and the reason we are doing the research in the first place. It's important not to treat user research as a silo-ed or singular activity independent of other research or insights that may exist to inform design and product strategy going forward. An important part of your role as a UX'er is to go out, question, and talk to other parts of an organization who can provide data to help, for example, market research, best practice, previous usability studies, expert opinion, focus groups, and so on. The more knowledge you can gather, the richer the story to tell people.

Choose the Right Method

Sometimes, we are far too protective over our toolset and we have seen practitioners try and pick the perfect method or process for a problem. Before starting the research, it's important to stand back and take the time to listen. Understand the questions that need answers and feel free to mix and match methods to get the answers. Also appreciate there WILL be gaps and you will not be able to answer all the questions with one round of research. "Rolling research" is key and to plan for constant insights should be just as important as the design, development and marketing efforts—it should form part of the business planning and product proposition. How do we know where we are today and the changes we need to make as we go forward? How do the insights help us get ahead that little bit more against the competition?

Slow Down and Speed Up and Slow Down Again

Product teams are moving faster to deliver and don't always take the time to stop and stand back to look at how they are delivering against past product lines or the competition. User research provides a reality check on the direction the product team is going in. It does not and should not provide all the answers. As researchers, we need to be very careful over promising the results of any research activity.

The Value of Asking Why

We should be questioning the value of what we make and its benefit to both the business and the people using the products and services. Often we see teams spending energies on proving the value of their own contributions (which is normal) instead of questioning the value of "the thing" itself. Some of the insights around value come from people's previous experiences, gut instincts, personal opinion, by speaking with other teams, and so on, but much of this can be informed through going out into the field to speak with users.

Dig Deeper and Then Dig Deeper Again

We should be able to look at a rich dataset from field research and squeeze all the goodness out of it to find the insight to bridge into design. For example: what are the takeaways for this workflow, screen, function, widget, copy, and so on? How does it help the users move forward in his or her journey towards completing a goal in both efficient and delightful ways?

Putting It All Together: The Bank

Recently we worked on a project for a bank to help them move from their existing website design to the next iteration of the design. The primary objective was to inform the new design with insights we were collecting through user research including usability reviews, competitor analysis, business and user discovery, design workshops and usability testing. It was really important to sit with the client up-front to understand what they were trying to achieve with the redesign effort and how we could compliment this in and around their timings. They were keen to show huge improvements from the current approach. Scope plays an important part of any research planning and you should also get a feel for the "research receptiveness" of the client as you plan the research beyond the first piece of work. What will work in Phase 1 and how does this inform Phase 2 and onwards?

Data Challenges

One of the challenges is how to take all the data you collect from the various research activities and make sense of it all. Also how you can take the team on the journey with you around their other work priorities so they feel they own the work instead of feeling like it's data being thrown over the fence with no place to go and no home.

Sell the Need

Sometimes we use terms for methods that our buyers do not understand, for example, Cognitive Task Analysis, GOMS, Hierarchical Task Analysis, Activity Theory, PARI (prediction, actions, results, interpretation), Syntactic Analysis, and so on. This is not to diminish the importance and value of the tasks, but the energy required to explain these can take away from the efforts required to better understand the need and then the approach we need to take.

"Talk Throughs"

One approach we have tried is "talk through."

So what are talk throughs?

During the user interviews, we would ask people to tell us a story on how they go about completing a task. We used the talk throughs to take into the design workshops as task flows to help drive the designs. The challenge was to get people to step outside of their domain (in this case "banking") and their own jargon and simply talk through a task, for example, "paying a bill." We would ask people do to this in both English and Chinese to listen to the steps they take and also the language or keywords they use. We also needed to be aware of creating an environment where people feel comfortable to tell a story that can lead to other stories and more insights. We took screen shots from competitors to overlay and see how it plays out next to the talk throughs, for example, What are the order of fields? What language is being used? What terminology can be simplified? What can be removed? What are we forcing users to do because of platform restrictions? What is the wish list to improve the experience? What are the gaps in our knowledge? Do we need to involve more users or more specialists to inform the design? For example, in one workshop we did not have a clear idea about the questions people have when buying and selling a specific investment product. So we quickly got on the phone and called a person who speaks directly to customers, is on the front line and sells products directly.

We also used the data from business discovery and user interviews to reduce our "guesswork" and to better understand:

- Product interaction: How users truly interact with the product, how much of the product they use, and what other software is used in the process
- Language: How users describe "products and services" to customers and the language they use
- Issues: Issues they face when using the software
- Tools: Other tools used to develop a customer solution. This can include brochures, fact sheets, pen and paper, and so on.
- Available time: How much available time the user has to deal with a task

Facilitating Design Workshops

Facilitating design workshops can be challenging at the best of times but especially so when the people attending do not speak English as a first language. It's important that you are comfortable with this as a facilitator and ensure you give the flexibility for people to speak in both languages (in this case English and Chinese). You should have someone in the room who can take over discussions in other languages and allow people to express their opinions, needs, issues and positives. A fluid discussion amongst different roles, skills, backgrounds and languages can build team spirit and push through improvements on the product. As trust builds, it's also key to allow for healthy arguments as the team continues on a journey towards something better.

Taking the "Talk Throughs" Forward into the Design

Talk throughs are simple flow diagrams drawn onto flip chart paper or a whiteboard to ground people back into what users want to achieve when completing a task. They navigate the team back to the core or essence of the task itself. This is important as discussions move forward and more layers of complexity are introduced including technology, politics, and visual design.

Takeaways

There is no substitute for immersing together with users to collect rich data to see how people use products and the issues they face to help drive the workflow of a new product design. Remember the following:

- Don't do user research for the sake of research—understand what you want to find out.
- Engineer your research to help bridge insights back into design.
- Respect other cultures, locales and language—allow people to express themselves in their language of choice.
- Don't reinvent the wheel—if you can inform design without doing in depth research, do it, but always question where research can help you dig deeper.
- Facilitate towards a goal—take what you have learned from research and facilitate it.

42.4 CASE STUDIES OF INNOVATIVE APPROACHES TO TASK ANALYSIS

42.4.1 Perspective Probe Case Study by Marianne Berkovich (Google)

Certain topics are considered taboo for polite conversation—sex, politics, money, and religion. But what if your team needs to understand one of these topics? We developed and used the Perspective Probe method to understand the sensitive topic of money and investing when we worked with the Google Finance team. Perspective Probe is a nonthreatening and whimsical way to get people's opinions on sensitive topics without asking them directly.

Think of the Perspective Probe as a way to co-prepare a delicious multicourse meal with your participants. You will provide the recipes, they will cook the dishes, and you will both enjoy them together. No one course is the "be all and end all" of the meal. It is even okay if one of the dishes does not turn out well. But taken together, it is quite a satisfying meal.

The Perspective Probe method consists of the following steps:

1. Creating activities targeted for what you want to learn, and then sending them to the participants
2. Having the participants complete the activities and send them back to the researcher
3. Conducting a debrief interview with each participant about the activities
4. Analyzing the results from all the participants and sharing the findings with the team

This case study describes how we went through these steps for the Google Finance project. The study was conducted over 5 weeks in the spring of 2008 with 11 participants in the United States.

42.4.1.1 Creating the Activities

The Google Finance team wanted to better understand how people think about investing, how they use their online portfolios as part of their other investment tools, so they could find opportunities to improve Google Finance. We first made a list of all the research questions that we had on the topic areas, then grouped and prioritized the questions. The pared down set of questions were the following:

- What items comprise participants' financial lives? What tools do they use?
- What are participants' financial goals?
- What is included and excluded from their portfolios?
- How do they currently use their online portfolios?
- What categories are important in their investments?
- How do they evaluate their portfolios?
- What else is important?

We decided that each research question would map to one activity that would be included in the Perspective Probe packet. Each activity would then shed light on one part of their perspective on investing. All the activities together would represent a complete picture of their standpoint.

As we brainstormed appropriate activities, we developed a set of guidelines to evaluate how well the activities fit our intended goal. The guidelines were the following:

1. *Simple*: Because we would not be there to explain the activity more fully than the instruction card provided with the activity, the activity had to be nearly self-explanatory.
2. *Open to interpretation*: We wanted to give the participants freedom to address the subtopic in the activity in a way that made the most sense to them, so we wanted the activity to leave room for their interpretation.
3. *Stand-alone*: So the activities could be done in any order.
4. *Different from each other*: To keep the participants interested and make sure they completed all the activities, each activity had to introduce something new.
5. *Fun and whimsical*: We were asking more time and effort from the participants than a usual study, so we wanted to make sure the experience was fun.

We piloted the activities with several Google employees to assess whether the instructions made sense and how long it would take participants to complete the activities. We then sent out the packets and asked participants to return the packets about a week after they received it.

42.4.1.2 Having the Participants Complete the Activities and Send Them Back to the Researcher

We recruited 16 participants across the United States who indicated that they used an online portfolio to manage their investments. Eleven participants ultimately completed the packet and the debrief interview. Each participant received a Perspective Probe packet containing a welcome letter, the activities, explanation cards with each activity, an inexpensive digital camera, and postage to mail the packet back.

The participants had about 1 week to complete the activities and mail back the packets. They generated 185 artifacts and photos.

42.4.1.3 Conducting Debrief Interview with Each Participant about the Activities

The following table lists the activity, research question it aimed to answer, and a brief description of what the participant was asked to do. (The numbers correspond to the question we listed earlier.)

Some participants responded with creativity to these activities. For example, one participant added her own twist and her own symbols in the "Draw Your Financial Life" activity. She drew each of her assets (such as 401k, life insurance, personal checking account, and her Volvo) as a hot air balloon lifting her up and each liability (such as children's education, household expenses, and consumer debt) as a stake tethering her to the ground. Another participant made simple collages instead of taking photos for the "Photo Checklist." During the interview, many participants said that they had fun working on the activities (Figure 42.6).

Activity	Description
1. Draw Your Financial Life	Participants were given an 11 × 17 piece of paper and instructed to draw the important items, tools, and flows that make up their financial lives.
2. Letter from the Future	Participants were asked to write a letter to their present selves from their future selves who had accomplished their financial goals, telling them what they have achieved and how they got there.
3. Portfolio Definition	Participants completed a dictionary definition based on their own view of what a portfolio was. We wanted to understand what was included and excluded from their portfolio.
4. Guided Tour of Your Portfolio	Participants filled a museum tour booklet of their portfolio and took photos of highlights on the tour. They covered private information with the provided sticky notes.
5. Teams of Investments	Participants completed a booklet of "teams" (groups) of investments that they had in their portfolios. They named each team and listed its members.
6. Portfolio Report Card	Participants listed "subjects" on which to evaluate their portfolios and then graded it.
7. Photo Checklist	Participants were given a checklist of open ended items to take pictures of to show their environments—such as where they keep track of investments, good sources of information about investing, and so on.

FIGURE 42.6 Activities and instruction cards in the Perspective Probe.

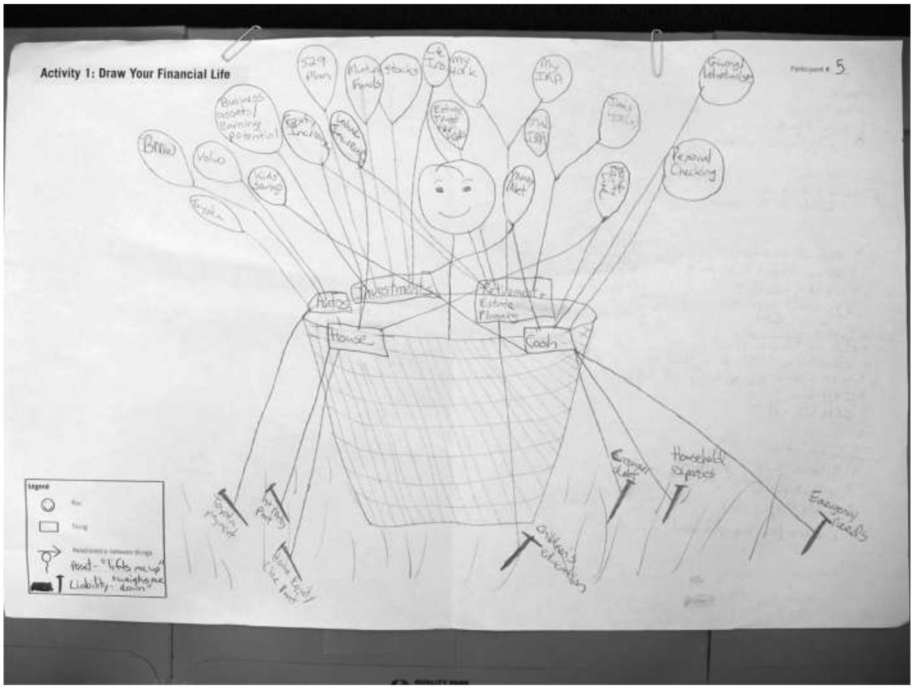


FIGURE 42.7 One participant's completed "Draw Your Financial Life" activity.

We found that the "Portfolio Definition" activity flopped and did not produce any interesting insights from participants. But one of the strengths of the Perspective Probe is that no one activity is the keystone. An activity can fail but the set of activities still yields good data.

During the one-on-one phone interviews, we used screen sharing to look at the digitized versions of their completed activities and discussed what they did. The sending of the packets back and forth had established rapport between the participants and researchers, so we were able to quickly get to substantive detail during the phone call (Figure 42.7).

42.4.1.4 Analyzing the Results from All the Participants and Sharing the Findings with the Team

We wanted to involve the whole team for the analysis of the findings, so we held a half-day workshop with engineers, product managers, interaction designers, and customer support staff. Before the workshop, each attendee was given a participant's packet and transcript of the phone call and asked to complete a one-page summary from a template we had prepared. During the workshop, we shared the one-page summaries with one another and noted insights, facts, and ideas on sticky notes. We created an affinity diagram with the sticky notes of themes that emerged.

One theme that emerged was that, at a high level, participants' financial concerns were similar, but the specific way that each participant dealt with the concerns varied greatly. For example, concerns we heard repeated were not needing to worry about finances in the future, making sure the portfolio is growing, having enough money for retirement or a child's education, not reacting to market fluctuations, being aware of short-term/long-term capital gains, and not having debt. However, we heard a variety of strategies about how to accomplish these in terms of how often portfolios should be checked and managed, as well as different definitions of "diversification" and different comfort levels with how diversified their portfolios were. After the workshop, we turned these and other insights into design guidelines for the interaction designers to refer to.

The Perspective Probe method helped us to collaborate with the participants to uncover rich data about their views on the sensitive topic of investing.

42.4.2 Triangulating Qualitative and Quantitative Methods for New Product Development by Jhilmil Jain (HP Labs*)

42.4.2.1 Introduction

This case study shows how bringing together (triangulating) information from qualitative and quantitative methods of understanding users, their needs, their tasks, and their scenarios helped us create a successful new product for small businesses.

42.4.2.2 Why Did We Do This Project?

At HP Labs, we have built a number of successful tools for individuals to manage personal projects such as planning a trip, working on an educational report, researching a product, and so on. Based on feedback from business customers, we identified the market opportunity to extend

* This project was conducted when the author was working at HP Labs.

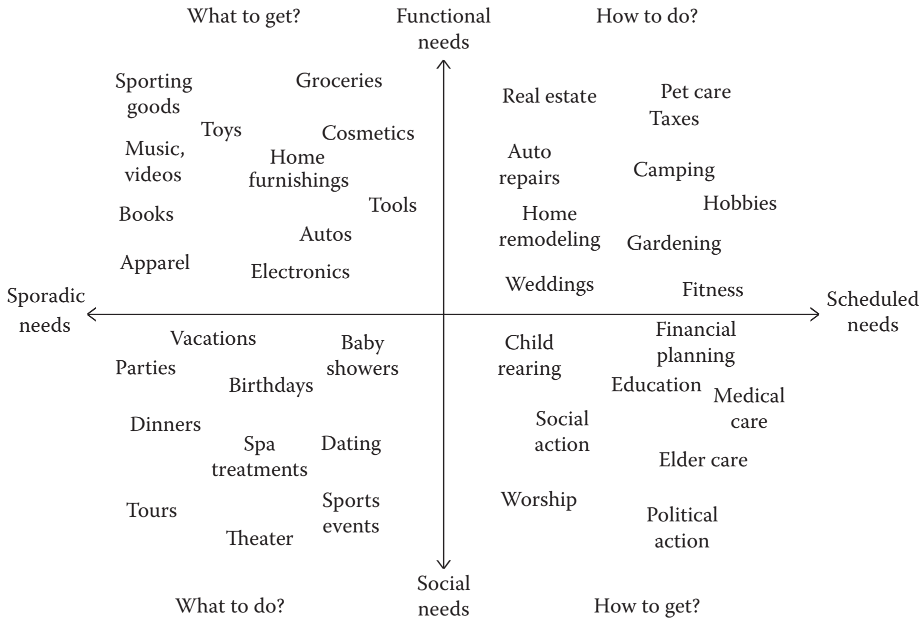


FIGURE 42.8 Project segmentation.

the "project-based" metaphor of these consumer-oriented tools to meet the needs of small companies that provide project-based services.

For many small business service industries (e.g., event planning, home remodeling, car customization, film production) managing vendor resources is a critical business function. Project managers and workgroups in these segments can be characterized by the following:

- Professional service businesses where project managers manage multiple complex projects simultaneously (e.g., a wedding planner plans several types of weddings and each could be in a different stage of planning).
- The majority of business activities involve sourcing* and managing a diverse set of interdependent vendors across 10+ different service and product categories (e.g., vendors such as caterers, florists, photographers, videographers, etc. need to often collaborate with the wedding planner).
- Projects require intensive collaborative client involvement to achieve highly customized solutions where clients have to make decisions with minimal expertise and time (e.g., the wedding planner works collaboratively with brides, grooms, and family members to plan the wedding).

The project managers/planners in these professional service industries are underserved by current vendor resource management software tools and platforms. The goal of our task analysis was to evaluate whether we could and should build a common cloud service-based platform† that provides a base level functionality to cater to multiple industries.

42.4.2.3 Which Task Analysis Methods Did We Use?

The first step was to identify specific industries to focus on in our task analysis. We conducted card-sorting as an internal-group exercise to identify relevant market segments that aligned well with HP's go-to market strategy for small businesses. (See Figure 42.8.) The x-axis of Figure 42.8 depicts projects across sporadic-scheduled needs, and the y-axis depicts projects across functional-social needs.

Based on our business goals and on the technologies that we wanted to incorporate in our new product, we identified industries that catered to the "scheduled and functional" needs of clients (top right quadrant), for example, event planning, home remodeling, video production, wedding planning, real estate, and so on.

First, we conducted 20 phone interviews with project planners and vendors and three focus groups with clients. These revealed the "day-in-the life-of" commonality across the various industries and we were able to generate a common experience map. (See Figure 42.9 for four domains such as event planning, wedding planning, video production, and home remodeling.) For example, in wedding planning, the project planner is the wedding planner, the client is the bride and groom, the vendors are people that provide products or services such as florists, caterers, photographers, and so on.

* Vendor sourcing consists of identifying customer needs, identifying vendors based on those needs, writing a request for proposal (RFP), soliciting input from vendors, analyzing results, creating contracts for the chosen vendors.

† Cloud platform services (also called "Platform as a Service [PaaS]") delivers a computing platform and/or solution as a service through the Internet. It makes it possible to use applications without the cost and complexity of buying and managing the underlying hardware and software. (See Cloud Computing Wikipedia [2010] for more details.)

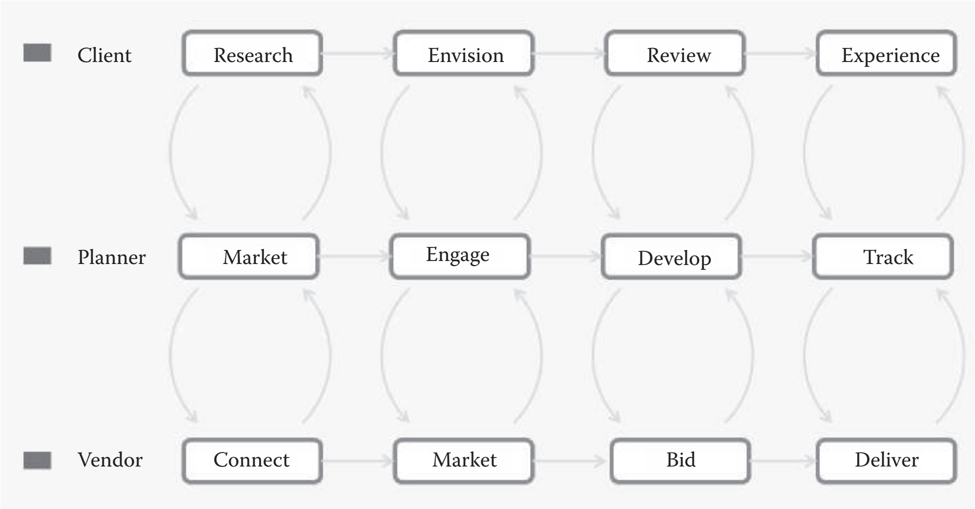


FIGURE 42.9 Experience map of three user types.

Second, we conducted five contextual observations (on-site office visits) to generate sequence and artifact models (see Figure 42.10) to depict the following:

- Detailed steps in the process flow for the planner to identify the workflow steps to be automated in the platform
- Digital versus physical artifacts used in each step such that we can identify the artifacts to be automated and tools to support in the automation; and also to identify the physical artifacts that are providing efficiencies and inefficiencies in the process
- Tools and software used such that we can evaluate which ones we need to integrate in the platform
- Stages where collaboration occurs with clients/vendors such that we can support this in the platform

Third, since we could not meet a number of planners, we designed and conducted two online surveys: to understand the needs of the planners and vendors. The survey was conducted using surveymonkey.com, where 56 wedding planners and 82 vendors took the survey and were given a $50 Starbucks gift card for their efforts. In the survey, we wanted to quantify the following:

- Service types/specialization offered
- Packages offered
- Fee structures for services and packages
- Marketing strategies and amount paid for them annually
- Software, physical artifacts, processed used for initial client engagement, project planning and management, finding and researching vendors, vendor proposals and estimates, contract management, client feedback and reviews
- Generic project planning software used (if any)
- Critical business pain points
- What are the roadblocks to business growth

By triangulating data collected from interviews, contextual inquiry, and surveys, we identified pain points, innovation opportunities, and design implications for each of the following features:

- Lead generation (e.g., mobile solution for entering leads)
- Initial meeting with client (e.g., provide standard, but customizable, forms; scheduling calendar)
- Client communication (e.g., currently all manual, but need access anytime, anywhere; integrate with e-mail)
- Vendor management (e.g., customizable categories, need to record contact info, pictures, brochures)
- Vendor bidding, final selection (e.g., very iterative and involves lots of communication, but client has no visibility)
- Vendor contracts (e.g., allow form upload; automation will be difficult due to lack of standardization)
- Payment management (e.g., mostly by checks, no need to integrate bank account)
- Reviews/feedback (e.g., used for testimonials for marketing and also to shortlist vendors)

The above information was provided as a video diary that not only served as a design inspiration but also helped designers, developers, and product managers experience the users' pain points.

We also created 10 personas with even distribution across various industries and user types with two key framework dimensions: (1) size of business (e.g., number of events planned per month) and (2) technical expertise and appetite of users.

Both the video diary and the personas turned out to be highly successful deliverables because designers and developers were not yet assigned to the project when we collected and analyzed this data.

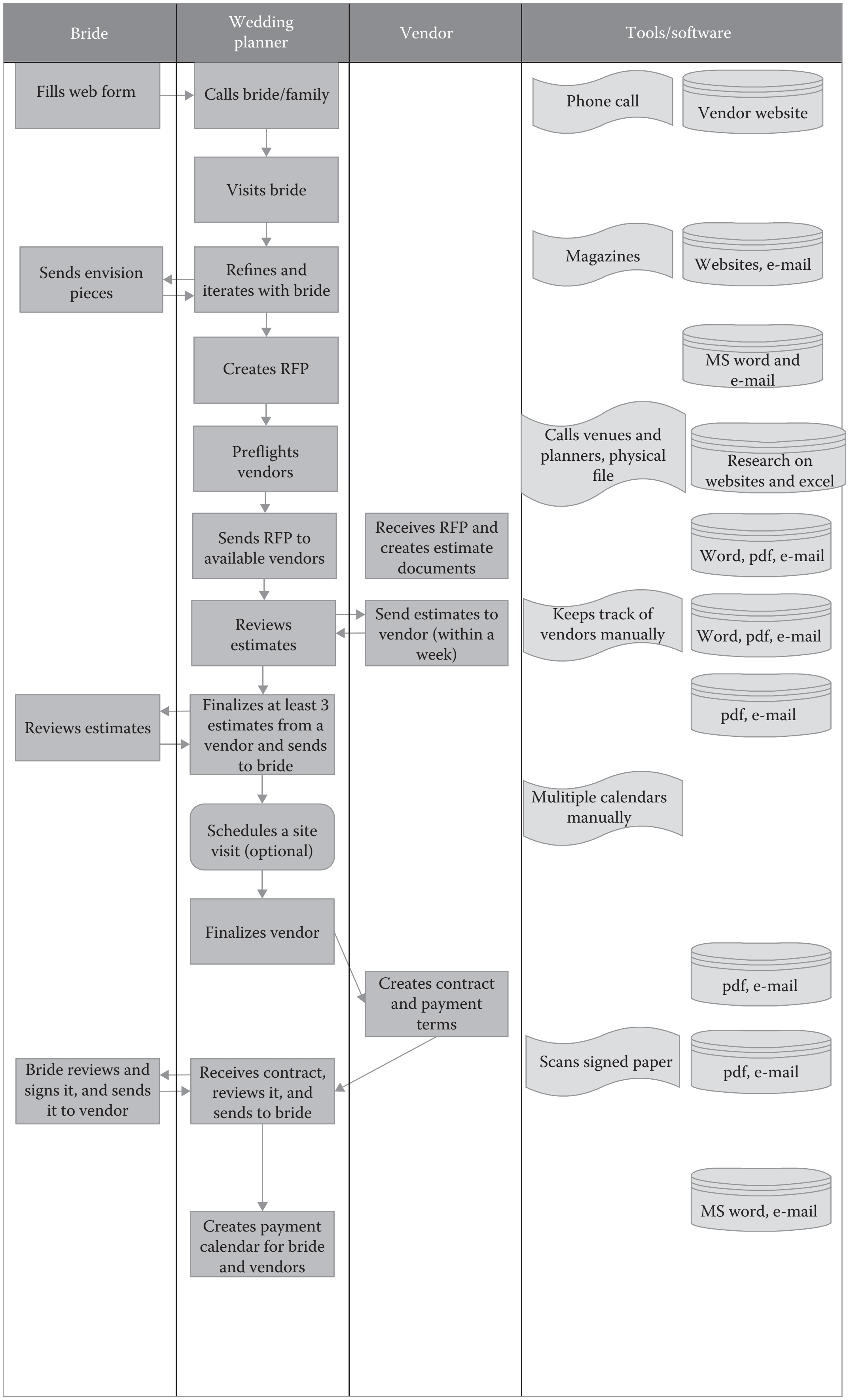

Bride
Wedding planner
Vendor
Tools/software
Fills web form
Calls bride/family
Phone call
Vendor website
Visits bride
Sends envision pieces
Refines and iterates with bride
Magazines
Websites, e-mail
Creates RFP
MS word and e-mail
Preflights vendors
Calls venues and planners, physical file
Research on websites and excel
Sends RFP to available vendors
Receives RFP and creates estimate documents
Word, pdf, e-mail
Reviews estimates
Send estimates to vendor (within a week)
Keeps track of vendors manually
Word, pdf, e-mail
Reviews estimates
Finalizes at least 3 estimates from a vendor and sends to bride
pdf, e-mail
Schedules a site visit (optional)
Mulitiple calendars manually
Finalizes vendor
Creates contract and payment terms
pdf, e-mail
Bride reviews and signs it, and sends it to vendor
Receives contract, reviews it, and sends to bride
Scans signed paper
pdf, e-mail
Creates payment calendar for bride and vendors
MS word, e-mail

FIGURE 42.10 Sequence and artifact diagram.

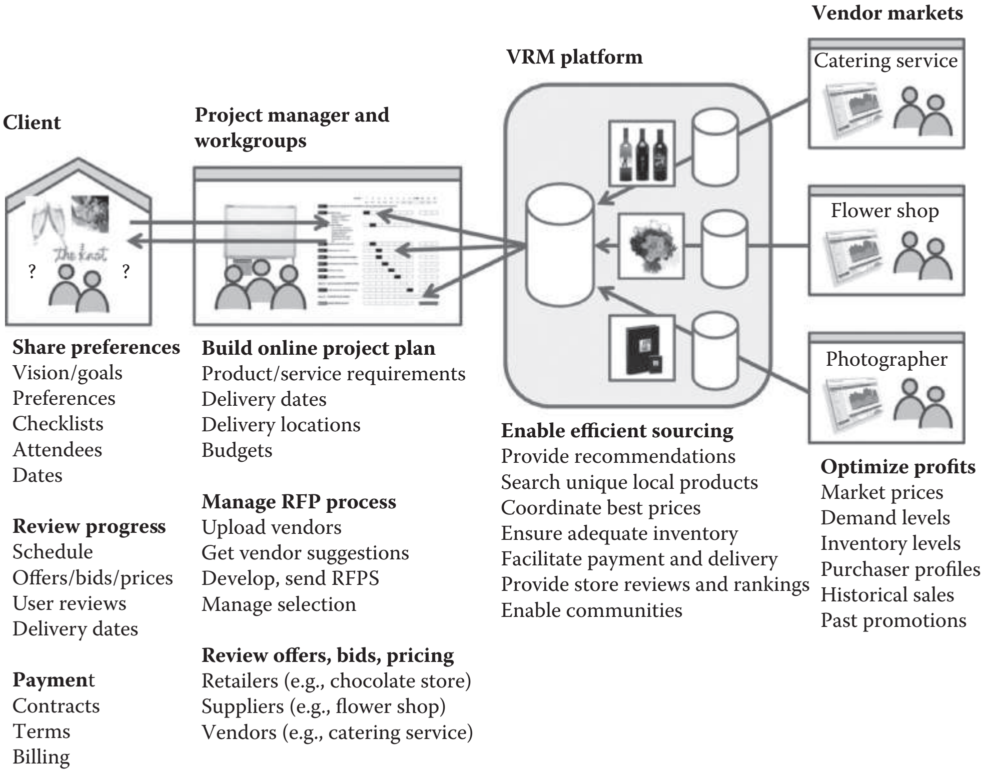


FIGURE 42.11 Overview of the Epicenter platform.

42.4.2.4 What Is Epicenter?

Based on the task analysis, we defined a brand new product called "Epicenter," which is a next-generation cloud service platform that connects small service companies with local suppliers/vendors and clients. (See Figure 42.11.)

- It is a web-based service that allows small service companies, such as wedding planners and interior architects, to more efficiently evaluate, source, and coordinate products and services required to implement their client's complex projects.
- It enables small suppliers and vendors to more efficiently market and supply products to local service companies by enabling them to have real-time visibility into the detailed project needs of small service companies and their clients.
- It enables clients to effectively create and communicate their requirements.

42.4.2.5 How Did We Handle the Need for Additional Task Analysis?

As we started designing each screen of Epicenter (see Figure 42.12), we realized that we needed to conduct additional task analysis for each stage in the workflow (requirement, RFP, proposal, contract, payment, feedback) for all three user types (planner, vendor, client) to help us identify the details of the process. Because we had limited resources at this stage, we chose to conduct rapid task analysis using the following steps:

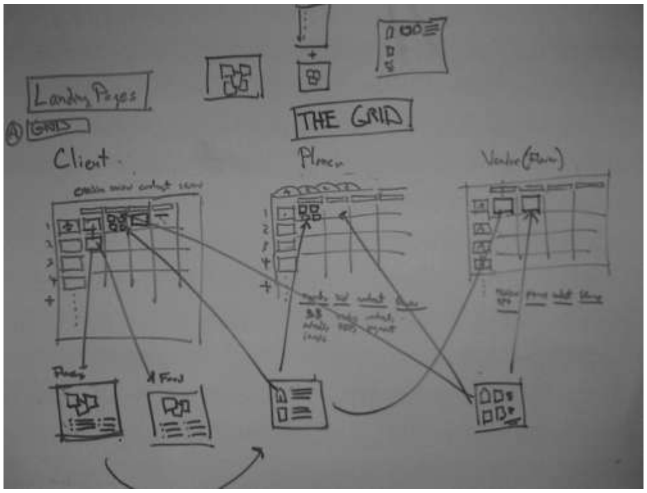


FIGURE 42.12 Initial Epicenter designs.

- We established a panel of three expert project managers (one each from event planning, video production, home remodeling) and we met with them every 2 weeks for a design review.
- We conducted scenario walkthroughs using low fidelity prototypes and focus groups with HP employees every month. (We were able to locate three women who were getting married/planning a wedding and a spouse who was a photographer.)

Additional task analysis not only helped us inform screen designs but also helped us prioritize features and functions.

42.4.3 Designing a Shared Media Experience: An Ethnographic Approach by Rama Vennelakanti (HP Labs)

42.4.3.1 Introduction

Unlike earlier times when television was an entertainment medium and the personal computer was a productivity device, the distinction and the boundaries are fuzzier today. Both are used to consume media as well as for productivity.

Today it is as familiar a sight to see a group of friends huddled around a laptop (or a desktop) checking out the latest posts on a photo sharing site or the latest episode of a popular soap as it is to see a group of friends or a family gathered in front of a TV to watch the World Cup or Wimbledon matches. A personal computer is used as much by a group to watch an online video as by an individual to produce an assignment; it moves between being a multiple-user device and a single-user device.

However what has not changed is the interaction mode—a single keyboard, a single mouse, a single remote. Both TVs and personal computers were designed for and have remained single user devices, at least as far as the interaction is concerned. So, what is otherwise a group experience/activity, such as watching videos or photos with friends, is often defined by the one person who has control of the interaction mode: the keyboard, mouse, and remote. This control over the interaction (or at least the desire for it) is often manifested in the tussle over the remote control. However, this single person interaction mode also ensures a sense of order by ensuring that the system receives just one command and control input at a time, thus avoiding chaos.

42.4.3.2 Case Study: What We Were Trying to Learn

The objective of the study was to identify patterns, if any, in the interactions that users have with media (and media devices) in an in-home group media-consumption scenario. Some of the questions we set out to answer were the following:

- What, if any, are the patterns of interactions?
- What are the factors that influence interactions with media?
- How do interactions change in a group situation?

42.4.3.3 How We Structured the Study

We recruited twelve users, across two cities, who met some basic criteria for ownership and use of media and media devices. They were given an incentive in the form of a gift or a gift coupon to participate in the study, apart from an occasional pizza for an activity with their friends or siblings.

As the focus was to identify parameters that influence interactions, we decided to study existing interaction behaviors, and to look for patterns in those behaviors. We were interested in users as they interacted with media individually and in groups, with media and media devices they are familiar with (their personal media and devices), with people they usually share these media with (family and friends), in surroundings they are familiar with (their homes), at times that were as close to usual for a particular media activity as possible.

42.4.3.4 Data: What We Collected and How We Collected It

Observation of interactions with each of these users was spread across 3–4 days depending on the media task. We spent a few hours on day 1 getting a profile of the user. We also took stock of the various devices in the user's home, specifications of those devices, and the user's familiarity and extent of use with those devices. We asked the users to categorize the devices based on ownership (even notional; i.e., the device was family owned or paid for by the parent but was used by the user), use, and media type among other parameters. We explored the user's media collections. We used a structured questionnaire method along with task flows to gather this data. (See Figure 42.13.)

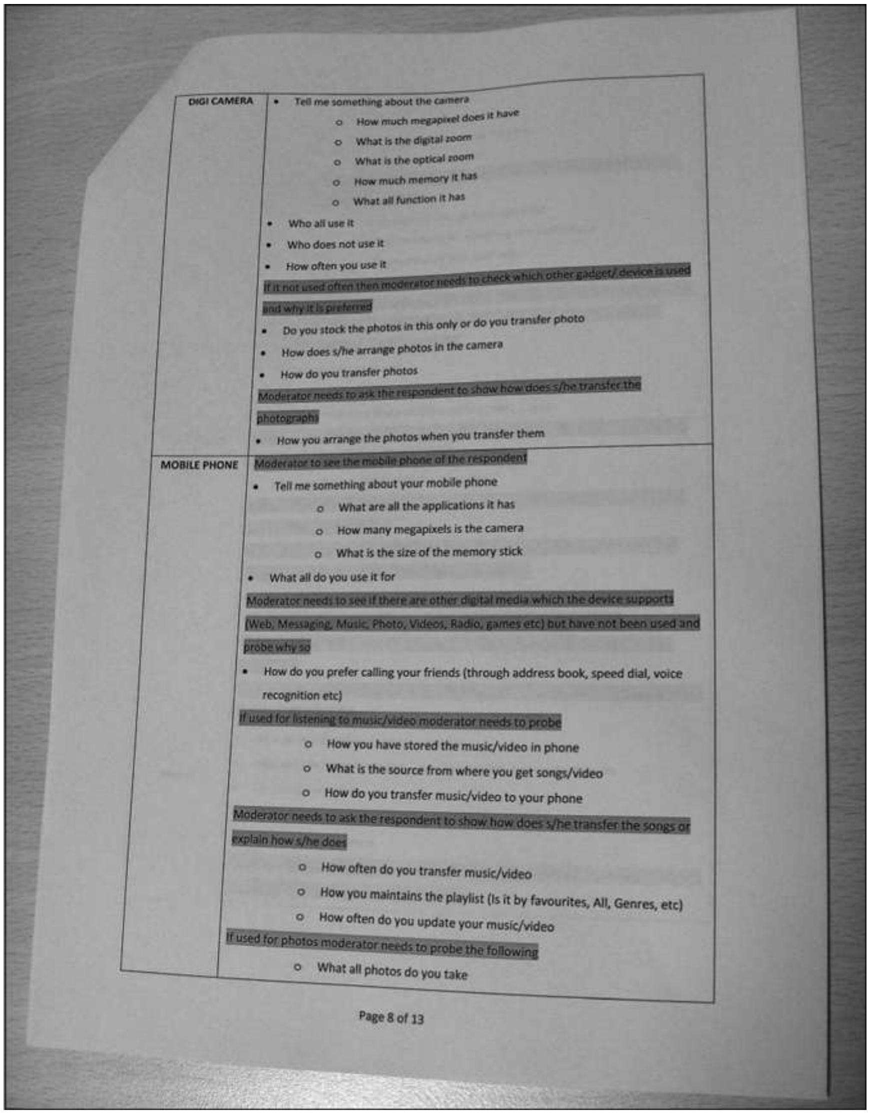
After this interaction on day 1, we set up specific activity slots, so that we could study the user's interactions with specific media types. The members for each activity (in a group activity) were decided by the user (as also the time of the day). We scheduled data collection at different times of the day at the convenience of the user. For single user activities, other members of the family—an inquisitive sibling or a parent joined in sometimes and this was allowed, as it provided us some natural interaction data.

We observed activities in various spaces in the home, where they normally happen. For example, if a family got together in a bedroom to watch a movie, then that became the observation scene. We observed the group activities passively; but single user activities were mostly interactive, with observations being followed by specific questioning to elicit information and understanding.

Some of the activities happened in their natural course and we observed them. For example, Sunday afternoon movie/video with the family was a regular happening, so we observed it as it happened. Some activities, such as a photo viewing session with friends, also happened in the natural course of events in the user's life. For example, some of the users had just returned from a trip and had arranged a photo viewing session before we got in touch with them; so we ended up observing this interaction. Some sessions were planned as part of the study where we asked the user to invite friends or cousins with whom they usually share a video game or watch the latest episode of "Heroes," and we observed the activity.

All activities were video recorded. We filled in observation sheets throughout the session. We had designed these observations sheets to capture information like seating pattern, the shape of the group (cluster), and so on, in drawings and notes. (See Figure 42.14.)

We took care to ensure that the video camera was unobtrusive. The observers always positioned themselves outside



DIGI CAMERA	• Tell me something about the camera ○ How much megapixel does it have ○ What is the digital zoom ○ What is the optical zoom ○ How much memory it has ○ What all function it has • Who all use it • Who does not use it • How often you use it If it not used often then moderator needs to check which other gadget/ device is used and why it is preferred • Do you stock the photos in this only or do you transfer photo • How does s/he arrange photos in the camera • How do you transfer photos Moderator needs to ask the respondent to show how does s/he transfer the photographs • How you arrange the photos when you transfer them
MOBILE PHONE	Moderator to see the mobile phone of the respondent • Tell me something about your mobile phone ○ What are all the applications it has ○ How many megapixels is the camera ○ What is the size of the memory stick • What all do you use it for Moderator needs to see if there are other digital media which the device supports (Web, Messaging, Music, Photo, Videos, Radio, games etc) but have not been used and probe why so • How do you prefer calling your friends (through address book, speed dial, voice recognition etc) If used for listening to music/video moderator needs to probe ○ How you have stored the music/video in phone ○ What is the source from where you get songs/video ○ How do you transfer music/video to your phone Moderator needs to ask the respondent to show how does s/he transfer the songs or explain how s/he does ○ How often do you transfer music/video ○ How you maintains the playlist (Is it by favourites, All, Genres, etc) ○ How often do you update your music/video If used for photos moderator needs to probe the following ○ What all photos do you take

Page 8 of 13

FIGURE 42.13 A sample of the structured questionnaire used to collect specs of the media device, their use, and possible tasks.

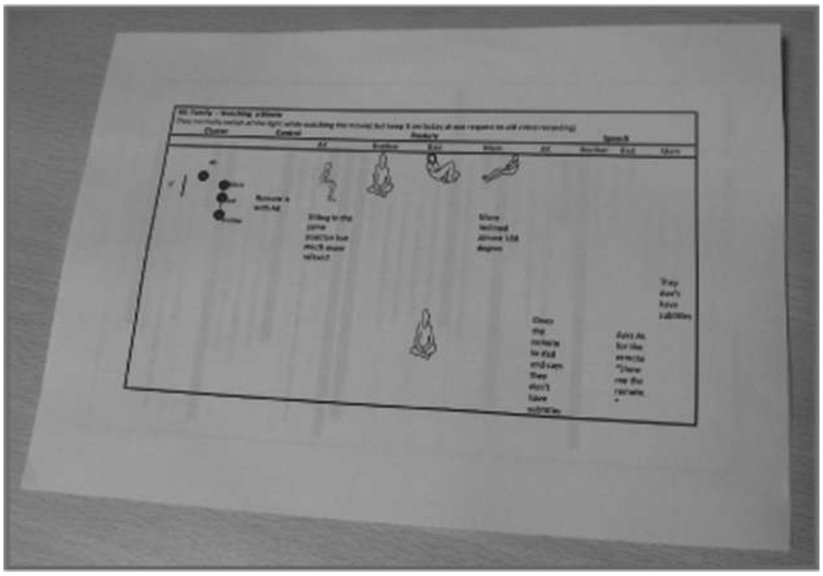
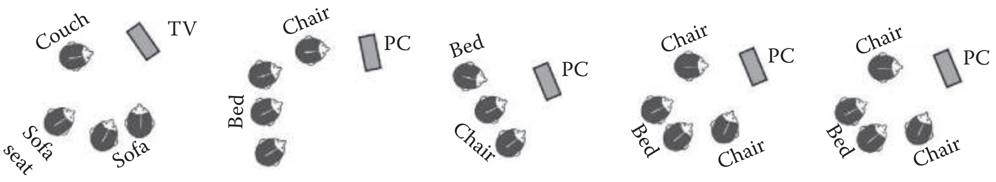


FIGURE 42.14 A filled up observation sheet.



Movie/serial on a TV

Movie on a personal computer

Photos on a personal computer

Gaming on a personal computer

Music on a personal computer

FIGURE 42.15 Depicting cluster shapes in the context of media—types and devices.

the activity itself, but they had a good view of the activity and users. We took still pictures at various points in time to illustrate specific observations.

With younger users (especially female) other family members were allowed to sit in through the interview sessions for as long as they liked and were even encouraged to look at the video camera monitor. Once they were convinced that the video recording and the interviews were very functional, they often left us alone and did not intrude into the exercise.

42.4.3.5 How We Analyzed the Data

We analyzed the video, voice recordings of the interactions and observations, along with observation notes and photographs of interactions for individual media types and media devices to identify interactions and factors that impacted them. We compared across media types and media devices to identify factors that impact interactions and the interaction patterns. We analyzed the data more than once and sifted through the data many times. Each round of analysis uncovered insights which led to looking for more patterns and so on both across and within media types and media devices. It was a painstakingly meticulous and a handcrafted manual exercise. We viewed the video tapes over and over again to look for patterns and confirm insights. We still continue to mine the analysis for patterns as our understanding of the area deepens.

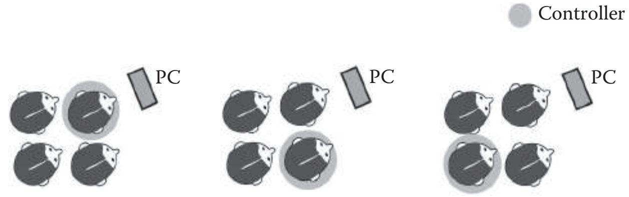
42.4.3.6 What We Learned

From our study, we found that various factors impact media interactions—media type, media device, the positioning/placement of the media device in the home, the members of the group and the power distance between them, the shape of the cluster formation of the group, time of day, media activity, role of the activity—primary (photo sharing), secondary (texting on the mobile phone while watching TV), or ambient (music while doing homework), and so on. (See Figure 42.15.)

All of these factors impact "control" of the media interactions (and the remote, mouse, and keyboard) and, therefore, the role of each person in the group. Control over the media and control over the media device meant control over interaction with the media.

42.4.3.7 Television

Control, in the context of interaction with the television, is driven by the remote control device. The remote is passed or taken/grabbed as a means of giving or taking control. This is done with ease.



Control with person close to computer

Control with person sitting next to computer

Control with person sitting behind

FIGURE 42.16 Shift of control among the group in the context of media consumption on a personal computer.

42.4.3.8 Computer

Possession of the keyboard and/or the mouse equals physical control over the interaction with the computer. Control over the keyboard and mouse tends to remain static through most of the activity, unlike the case of the television, where the remote can easily change hands and thus control can also change. Static control, as in the case of a personal computer, could be due to the fact that today using the computer is a sit-down activity. It is more difficult to dislodge a sitting user than to pass the remote to another person, as in case of the TV. However, at the computer, although the person carrying out an instruction or a command (that is, the person in possession of the keyboard/mouse) may not change, the person who is actually in control may. That is, the person who gives the command/instruction, and so controls the interactions, may be different from the person who controls the interaction devices. This brings forth another role a group member may play at the computer: the executor, the one who executes commands that are issued by someone else. Control itself (who decides what to do) may shift even if the executor (who uses the keyboard/mouse) does not. (See Figure 42.16.)

42.4.3.9 Challenges Our Insights Bring to Design

Existing behavior patterns provide us with significant insights into the research challenges that need to be addressed by technology research and user interface design to provide for a rich user experience. From our on-site observations and data collection and the insights from our data analysis, we can see the very interesting challenges that need to be addressed while designing intuitive interactions for accessory (keyboard, mouse, and remote) free group interactions in the context of media consumption.

Key design challenges include how to

- Pass and share control
- Allow interactions for all
- Develop rules for handling multiple and conflicting command and control inputs to the device to avoid chaos

Another key challenge is when and how much to actually "design" and when to let the social dynamics of the interactions take charge. Overdesigning may lead to a completely controlled experience which may shift focus of the experience from media consumption to control, which may end by being user unfriendly.

42.4.4 Making Task Analysis Tangible with Mental Models by Indi Young

The mental models I develop are simple affinity diagrams. They are constructs that lay out the degree to which your organization is supporting what people want to accomplish. Mental models not only include the steps people go through, but also the philosophies people enact and the emotional reactions people experience during the process. A range of knowledge like this allows you to empathize with different groups and see things from their perspectives (Figure 42.17).

Aligned beneath the affinity groups of these steps, emotions and beliefs are the ways in which an organization enables people to understand, manipulate, achieve, or affect things. It also might show how an organization helps alleviate frustrations or removes workarounds. This alignment provides a foundation from which you can map out a path to improve what your organization does for people, whether that means improving services you already provide, redesigning artifacts, or brainstorming new approaches based on an empathic viewpoint.

In his book, *The Design of Everyday Things*, right after discussing his example of the refrigerator/freezer controls, Don Norman writes this definition of mental models: "the models people have of themselves, others, the environment, and the things with which they interact" (page 17 in the first Doubleday/Currency edition, 1990). The mental models I draw are not about things you interact with; they are about the "others" in this definition. They are diagrams your team uses to explore the behavior of certain audience segments. I could have called them "alignment diagrams" or "empathy models," but imagine how a business manager would react to spending budget on "empathy." "Don't we get empathy from all the demographic marketing surveys?" (Surveys are not conversations. http://www.rosenfeldmedia.com/books/mental-models/blog/oxymoron_scientific_survey/) I had to use a name like "mental models" that business leaders could get behind so that the needed information could get gathered.

Mental models do not change very quickly; therefore, they are helpful as long-range road maps for organizations. Imagine the mental model of how a person obtains ordinary groceries. Currently that model has to do with supermarkets, perhaps with farmers markets and veggie boxes thrown in. One hundred years ago, before the advent of supermarkets, that model would have consisted of relationships with specialty vendors, like the milkman and the corner produce shop. Five hundred years ago, depending on where on the globe you lived, the mental model probably revolved around a weekly town market. The philosophies and behaviors around getting groceries change slowly over the generations. Thus, an organization can rely on a mental model during its lifespan, focusing on supporting the behaviors in better ways, while also keeping an eye out for slowly drifting changes.

Mental models help your organization create relevant solutions for people. In a *Tea with Teresa* podcast (http://www.teawithteresa.com/podcasts), researcher Natash Alani talks about her research into nonliterate people in the Kutch region of India who use mobile phones. She discovered that the people she was talking to would not even use the keypad to enter digits, and the connection between the screen and the keypad was hazy in their minds. What was sharply in focus was their desire to hear their daughter's voice on the phone or to talk to their grandparents. In her research, Natasha found that when these people took public transit, they could not read bus schedules. To find out when the bus was arriving, they would ask a nearby fruit stand vendor if a bus had just passed through or not. This local behavior illustrates the importance of establishing relationships and communicating with associates around them. Later Natasha felt troubled that designers did not understand the situation well enough when she heard a suggestion for putting an electronic message marquis up to show times when the next bus would come. That solution would take away the opportunity for a person to talk

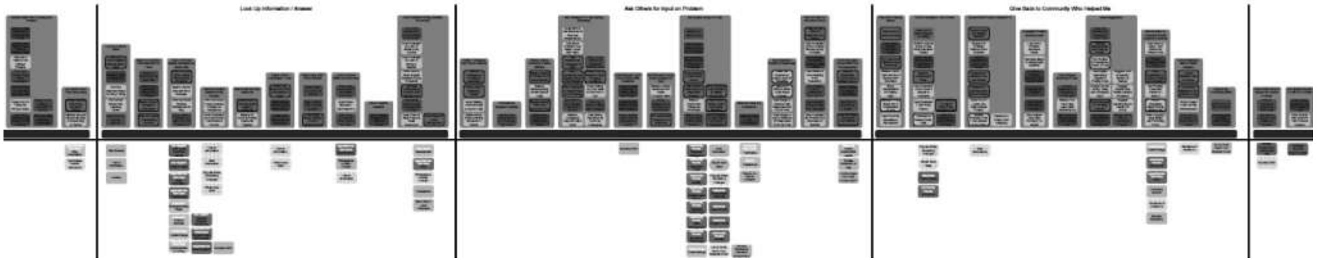

Look Up Information / Answer
Ask Others for Input on Problem
Give Back to Community Who Helped Me

FIGURE 42.17 An example mental model. Upper towers represent behaviors, philosophies, and reactions. Lower boxes aligned with towers show where your organization supports what people are doing.

to that fruit stand vendor. Instead, Natasha suggests, "Allow the fruit stand vendor to talk to the bus driver directly and find out if he is caught in traffic or will be arriving shortly." This action supports the known culture and empathizes with the way people place importance on the chance to chat with others around them. It is culturally relevant. And culturally relevant solutions are more effective—more *used*—and are more comfortable, learnable, and successful than solutions that ignore the behaviors of people in situ.

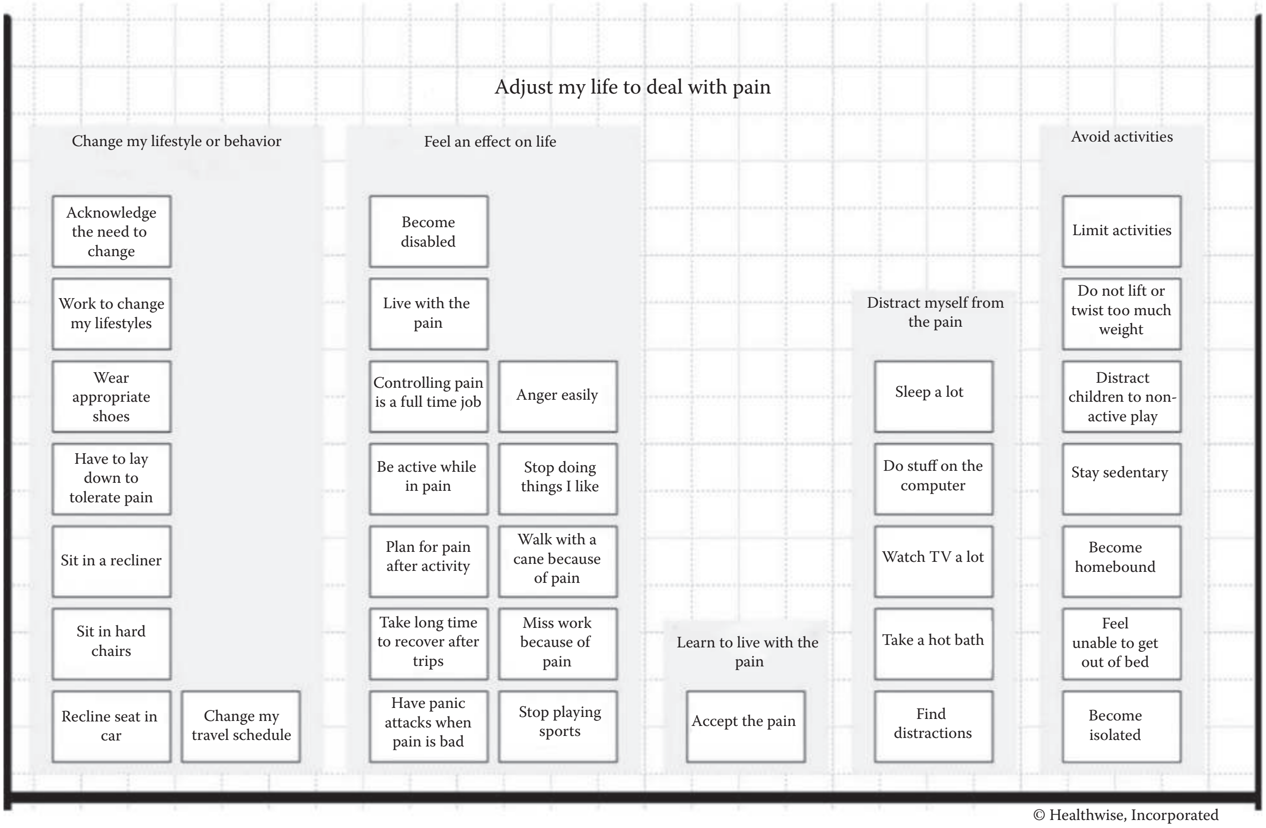
42.4.4.1 Case Study: Transforming Company Thinking

Healthwise (www.healthwise.org) is a nonprofit organization with a mission to help people make better health decisions. Healthwise works with health plans, hospitals, disease management companies, and health websites to provide the most current, scientific, evidence-based, consumer-friendly information to those who need it—encouraging patients to participate in decisions about their care, encouraging them to take action about their health, and offering support for the sometimes hurried and unclear synopsis doctors provide. The company began in 1975, offering printed decision-support and self-care materials. In the mid-1990s, Healthwise started creating an online resource of health information that would cover virtually all of medicine. Today the Healthwise® Knowledgebase includes more than 8000 in-depth topics and appears on many hundreds of websites.

Recently, Healthwise realized that the information it was providing could be more effective with improvements in search and browse functions. The user experience team within Healthwise spearheaded an initiative to improve information architecture for the knowledgebase and tools. But, with all the different ways of looking at the data, what was the best approach to create a single information architecture? At the same time, Healthwise wanted to expand its offerings. It wanted to offer more specific decision-support solutions for certain situations—but which situations? And what kind of support would be truly useful to the populace?

42.4.4.2 Starting with One Site

To answer these questions, Julie Cabinaw, Director of User Experience, decided to create a series of mental models to derive information architecture and to verify which of the many directions Healthwise had in mind to follow. However, first she wanted senior management to agree that this kind of research would be a good return on investment. She decided the most persuasive evidence would be a successful micro-site that incorporates knowledge found in the first mental model they would make. A micro-site is a site devoted to one health condition, like high blood pressure or diabetes, and does not expect the audience to identify itself—in other words, users remain anonymous. Her team spent one month creating a mental model about dealing with low back pain (Figure 42.18). The team learned some areas that were


Adjust my life to deal with pain
Change my lifestyle or behavior
Acknowledge the need to change
Work to change my lifestyles
Wear appropriate shoes
Have to lay down to tolerate pain
Sit in a recliner
Sit in hard chairs
Recline seat in car
Change my travel schedule
Feel an effect on life
Become disabled
Live with the pain
Controlling pain is a full time job
Anger easily
Be active while in pain
Stop doing things I like
Plan for pain after activity
Walk with a cane because of pain
Take long time to recover after trips
Miss work because of pain
Have panic attacks when pain is bad
Stop playing sports
Learn to live with the pain
Accept the pain
Distract myself from the pain
Sleep a lot
Do stuff on the computer
Watch TV a lot
Take a hot bath
Find distractions
Avoid activities
Limit activities
Do not lift or twist too much weight
Distract children to non-active play
Stay sedentary
Become homebound
Feel unable to get out of bed
Become isolated

© Healthwise, Incorporated

FIGURE 42.18 Part of the low back pain mental model. Shows upper part of diagram where people talk about how they change the way they live because of this pain.

important to people with back pain where Healthwise could improve content, such as around how to deal with chronic back pain. They also learned that the use of alternative medical treatments, such as chiropractic or acupuncture, was very important to many people with low back pain. Through the mental model, they also uncovered how important having support—through family, friends and caregivers—was to people with low back pain, and how difficult it was for people in pain to ask for the help they need. These new insights proved the value of this kind of research.

To build the mental model, the team went through a few phases: they set the scope for their research, recruited participants for the research, conducted interviews, analyzed the data, and then built the mental model. In order to ensure contextual and appropriate information, the team set a simple but defined scope for the interviews, which was "to gain insight into how people understand and manage their back pain." The next step was to carefully recruit participants for the interviews to ensure a mix of people dealing with different back pain situations, across a variety of demographic backgrounds, such as age, gender, race, education and income levels. The team conducted nondirected interviews in which the interviewer lets the interviewee guide the conversation, picking up on clues provided to dive deeper into all possible ways that the person may think about the subject. Curiosity is the key to these interviews. From these interviews, the team analyzed the data to understand people's tasks, philosophies and feelings about dealing with back pain. They used spreadsheets to organize and group themes and patterns based on verbs, from which they built out the mental model. Finally, the team used the mental model about low back pain to understand the major arenas of people's thinking. They found mental spaces such as *understand cause of my back pain*, *treat my back pain* and *adjust my life to deal with the pain*. Since the hierarchical design of the model lends itself neatly to information architecture, the user experience team was able to create naturally understandable navigation for the Healthwise information and tools, such as Symptoms & Causes, Treatment, and Living & Coping.

The micro-site content derived from this mental model has delivered on expectations. Users visiting the pilot site at a major health plan are finding the content they need and rating the content they review as valuable to them in managing their condition. The mental model diagram allowed Healthwise to see where they could support people better and where there were new areas that conventional thinking about low back pain health care did not cover (Figure 42.19).

42.4.4.3 Expanding to the Rest of the Company

Mental models have spread to the rest of the company. Healthwise sales folks explain the mental model process when they present the company's solutions, so that clients understand that the information architecture comes from real people and real stories. Executives have been open to expanding the ways to support one of the core philosophies that has guided the company, evidence-based information, because of what mental model research has uncovered. Not all information and tools that the layperson is after are based on traditional medicine. For example, the mental model about low back pain revealed the importance of chiropractors and acupuncture. In response, the Chief Medical Officer and his team expanded relationships with experts in these areas to help validate and improve the quality of information for complementary and alternative medicine, such as chiropractic and acupuncture, to people suffering back pain.

> Mental models are emphatically part of the DNA of Healthwise now," says Julie. Her user experience team has created mental models about wellness, coping with a health condition, and pregnancy. Healthwise plans to create products supporting people with these health interests. Julie notes, "Now product managers come to me asking us to create a mental model for a product. Two years ago, that wasn't even in our company vocabulary.

For more on the Mental Models approach to task analysis, see Young (2008).

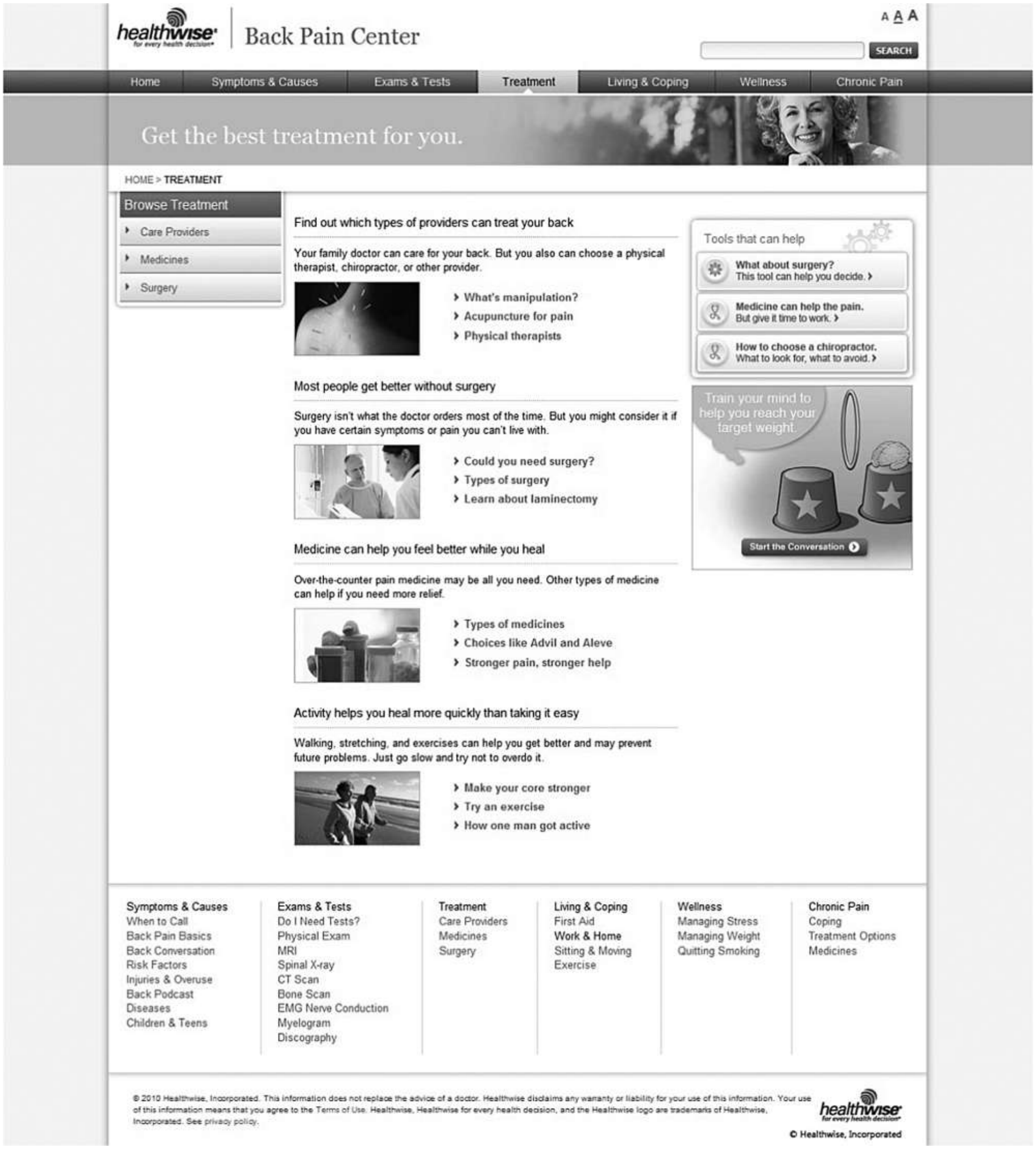
42.5 CONCLUSION

Many usability practitioners do task analysis in one form or another, although often they do not recognize it as such, much as Moliere's character found the fact that he had been speaking prose all his life a revelation.

Some may hesitate to plan for task analysis because they think of it as too complex and time consuming to apply in real-world situations where time is always too short. In this chapter, we have tried to show that task analysis is a family of flexible and scalable processes that can fit well in almost any development environment.

Some may have thought of task analysis only in terms of highly structured ways of capturing minute details of specific procedures relevant to evaluating already created designs or developing training or documentation for already determined systems. In this chapter, we have tried to show that while such uses of task analysis continue, the more common use today is in developing a very broad understanding of users' work and is of great use from the earliest strategic planning stages through all the phases of predesign. Task analysis as laid out in this chapter continues to be useful throughout the process, being used at later stages to develop scenarios for user-oriented evaluations as well as in inspection methods.

Task analysis is a way to involve the entire team in understanding users. It provides ways to organize the mountain of unstructured data that often comes from field studies or site visits. It is an essential part of the process of creating any product (software, hardware, website, document) because products are tools for users to accomplish goals; products are all about doing tasks.



Back Pain Center

Get the best treatment for you.

HOME > TREATMENT

Browse Treatment
- Care Providers
- Medicines
- Surgery

Find out which types of providers can treat your back

Your family doctor can care for your back. But you also can choose a physical therapist, chiropractor, or other provider.

- What's manipulation?
- Acupuncture for pain
- Physical therapists

Most people get better without surgery

Surgery isn't what the doctor orders most of the time. But you might consider it if you have certain symptoms or pain you can't live with.

- Could you need surgery?
- Types of surgery
- Learn about laminectomy

Medicine can help you feel better while you heal

Over-the-counter pain medicine may be all you need. Other types of medicine can help if you need more relief.

- Types of medicines
- Choices like Advil and Aleve
- Stronger pain, stronger help

Activity helps you heal more quickly than taking it easy

Walking, stretching, and exercises can help you get better and may prevent future problems. Just go slow and try not to overdo it.

- Make your core stronger
- Try an exercise
- How one man got active

Tools that can help

What about surgery? This tool can help you decide. ›

Medicine can help the pain. But give it time to work. ›

How to choose a chiropractor. What to look for, what to avoid. ›

Train your mind to help you reach your target weight.

Start the Conversation

Symptoms & Causes
When to Call
Back Pain Basics
Back Conversation
Risk Factors
Injuries & Overuse
Back Podcast
Diseases
Children & Teens

Exams & Tests
Do I Need Tests?
Physical Exam
MRI
Spinal X-ray
CT Scan
Bone Scan
EMG Nerve Conduction
Myelogram
Discography

Treatment
Care Providers
Medicines
Surgery

Living & Coping
First Aid
Work & Home
Sitting & Moving
Exercise

Wellness
Managing Stress
Managing Weight
Quitting Smoking

Chronic Pain
Coping
Treatment Options
Medicines

© 2010 Healthwise, Incorporated. This information does not replace the advice of a doctor. Healthwise disclaims any warranty or liability for your use of this information. Your use of this information means that you agree to the Terms of Use. Healthwise, Healthwise for every health decision, and the Healthwise logo are trademarks of Healthwise, Incorporated. See privacy policy.

© Healthwise, Incorporated

FIGURE 42.19 Example from low back pain micro-site. Healthwise added sections for manipulation and acupuncture to support what they saw in the mental model.

REFERENCES

Ann, E. 2004. Cultural differences affecting user research methods in China. In *Understanding Your Users*, ed. C. Courage and K. Baxter, 196–207. San Francisco, CA: Morgan Kaufmann.

Butler, M. B. 1996. Getting to know your users: Usability roundtables at Lotus Development. *Interactions* 3(1):23–30.

Butler, M. B., and M. Tahir. 1996. Bringing the users' work to us: Usability roundtables of Lotus development. In *Field Methods Casebook for Software Design*, ed. D. Wixon and J. Ramey, 249–67. New York: Wiley.

Chavan, A. 2005. Another culture, another method. In *From the Proceedings of the Human Computer Interaction International Conference*. Las Vegas, NV. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Cloud Computing Wikipedia. 2010. http://en.wikipedia.org/wiki/Cloud_computing.

Cooper, A. 1999. *The Inmates Are Running The Asylum*. New York: Macmillian.

Courage, C., and B. Baxter. 2004. *Understanding Your Users.* San Francisco, CA: Morgan Kaufmann.

Courage, C., G. Redish, D. Wixon. 2007. Task analysis. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, (Human Factors and Ergonomics)*, Second ed., ed. A. Sears and J. Jacko, 927–48. Mahwah, NJ: Lawrence Erlbaum Associates.

Dray, S., and D. Mrazek. 1996. A day in the life of a family: An international ethnographic study. In *Field Methods Casebook for Software Design*, ed. D. Wixon and J. Ramey, 145–56. New York: Wiley.

Dray, S., and D. Siegel. 2005. Sunday in Shanghai, Monday in Madrid? In *Usability and Internationalization of Information Technology*, ed. N. Aykin, 189–212. Mahwah, NJ: Lawrence Erlbaum Associates.

Flanagan, J. C. 1954. The critical incident technique. *Psychol Bull* 51(4):327–58.

Foucault, B. 2005. Contextualizing cultures for the commercial world: Techniques for presenting field research in business environments. In *Proceeding of the HCII Conference*. Las Vegas, NV.

Foucault, B., R. Russell, and G. Bell. 2004. Techniques for researching and redesigning global products in an unstable world. A case study. In *Proceedings of CHI2004 Conference on Human Factors in Computing Systems.* New York: ACM Press.

Gray, W. D., B. E. John, and M. E. Atwood. 1993. Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world performance. *Hum Comput Interact* 8:237–309.

Hackos, J. T., and J. C. Redish. 1998. *User and Task Analysis for Interface Design*. New York: Wiley.

Kirwan, B., and L. K. Ainsworth. 1992. *A Guide to Task Analysis*. London: Taylor & Francis.

Kujala, S., M. Kauppinen, and S. Rekola. 2001. Bridging the gap between user needs and user requirements. In *Proceedings of PC-HCI 2001 Conference*, Patras, Greece.

Lee, W. O., and N. Mikkelson. 2000. Incorporating user archetypes into scenario-based design. In *Proceedings of the Ninth Annual Conference UPA 2000*. Chicago, CA: Usability Professionals' Association. www.upassoc.org.

Lovejoy, T., and N. Steele. 2005. Incorporating international field research into software product design. In *From the Proceedings of the Human Computer Interaction International Conference*. Las Vegas, NV. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Miles, M. B., and A. M. Huberman. 1994. *Qualitative Data Analysis: An Expanded Source Book.* New York: Sage.

Mulder, S., and Z. Yaar. 2007. *The User is Always Right*. Berkeley, CA: New Riders.

Norman, D. 1988. *The Design of Everyday Things*. New York: Doubleday. (Originally published as *The Psychology of Everyday Things*; hard cover published by Basic Books).

Pruitt, J., and T. Adlin. 2005. *The Persona Lifecycle*. San Francisco, CA: Morgan Kaufmann, an imprint of Elsevier.

Pruitt, J., and T. Adlin. 2010. *The Essential Personal Lifecycle, (Abridged Edition of the 2005 Book)*, Burlington, MA: Morgan Kaufmann, an imprint of Elsevier.

Quesenbery, W., and K. Brooks. 2010. *Storytelling for User Experience*. Brooklyn, NY: Rosenfeld Media.

Redish, J., and D. Wixon. 2003. Task analysis. In *The Human-Computer Interaction Handbook*, ed. J. Jacko and A. Sears, 922–40. Mahwah, NJ: Lawrence Erlbaum Associates.

Siegel, D., and S. Dray. 2005. Making the business case of international user centered design. In *Cost-Justifying Usability: An Update for the Internet Age*, ed. R. Bias and D. Mayhew. San Francisco, CA: Morgan Kaufmann.

Simpson, K. T. 1998. The UI war room and design prism: A user interface design approach from multiple perspectives. In *User Interface Design: Bridging the Gap from User Requirements to Design*, ed. L. Wood, 245–74. Boca Raton, FL: CRC Press.

Wixon, D. 1995. Qualitative research methods in design and development. *Interactions* 2:19–24.

Young, I. 2008. *Mental Models: Aligning Design Strategy with Human Behavior*. Brooklyn, NY: Rosenfeld Media.

43 Contextual Design

Karen Holtzblatt

CONTENTS

43.1 INTRODUCTION

Contextual Design (CD) is a user-centered design process that takes a cross-functional team from collecting data about users in the field, through interpretation and consolidation of that data, to the design of product concepts and a tested product structure. CD has been used in companies and taught in universities all over the world, along with other user-centered design processes. This chapter introduces the steps of CD; for a full description, see *Contextual Design*[1] and *Rapid Contextual Design*.[2]

Over the last 20 years or more, the high-tech industry has moved from using an engineering-driven requirements and design process to more user-centered processes. It has moved away from tail-end usability testing of products conceived by collecting features from customers, users, internal stakeholders, and enhancement databases. Although, not all companies practice user-centered design, most know that achieving best-in-class solutions depends on designing from a deep understanding of what people do, so that products enhance the activities of daily life to make it more efficient and delightful. And in the last 20 years, product managers and designers alike have come to understand that a "deep understanding" of what customers or users do means collecting and working with field data.

Engineering-driven design can be summarized succinctly as, "putting 10 smart guys in a room to decide what might be 'cool' technology to ship." The ultimate challenge for organizations is that understanding the customer is hard, takes time, and challenges entering assumptions and job activities. The alternatives—talking to each other rather than users, building internal organizational buy-in for the next release, looking at surveys, analyzing bug and enhancement databases, and creating rapid high-fidelity prototypes, and showing the ideas to people in a focus group or on the web—feel easier and seem like a valid way to gather requirements. They also do not challenge people who are used to interact with tools to do social science type requirements work.

But as platforms evolved and evolved more, and as we moved from natural-language command line interfaces to WYSIWYG to VisiCalc to the web to the iPhone, "regular" people became the real customers. And like it or not, engineers, product managers, and user experience designers are not regular people. They love technology too much. Their instincts are not in sync with the lives of real people. Their willingness to mess with technology is not in sync with real people. So, if teams are going to design successful products for real people, they need to get exposed to and use reliable data about the lives of the people they design for.

Field data with a process to use it for design is even more important today than in the past. Why? Technology is filtering into every nook and cranny of people's lives. Twenty years ago, we were only just introducing the idea of understanding the lives of real people to companies making productivity software and tools for a market. Then, enterprise software companies became aware of the power of field data for driving new concepts for their products and ensuring their products really worked for people. Recently, companies making hardware/software products like mobile phones, medical devices, and measuring instruments have also started to become aware of the need for field data. All of these companies make products for sale. They need to pay attention to their customers or their products will not be bought. They are motivated to find ways to ensure that what they make will enhance and delight customers to be successful—and they need to deliver on time within an engineering process.

The value of field data for getting any technology system right has been discovered over and over by industry after industry. Recently, IT and businesses have started to recognize the power of field data. There are some early adopters, but only this year analysts are starting to talk about a user-centered model for business systems.[3] These same companies earlier recognized that they need data for their customer-facing websites, but that is not the same as using powerful data to help drive the internal systems that support the business itself.

Today, technology has infiltrated every aspect of the work of employees and the process redesign implies simultaneous process and technology redesign. More companies are realizing that understanding the employees as user is also a part of choosing and customizing third party systems. At the same time, these businesses are challenged by the changing expectations that employees bring from their personal life—web surfing, social networking, blogging, and lightning-fast finding that produces the right information. Figuring out what systems should be like for these workers is a big challenge requiring a deep understanding of both the business needs and the practices of the people.

On the consumer side we are in the middle of a serious business revolution. Technology is becoming a part of our cars and our homes. Utility companies want to reach into our houses to turn down our thermostats. Technology is challenging many traditional industries to look for new business models and products: publishing, music, retail, banking, and a myriad of other services. These companies are not used to designing their products as software products or web-based services. These companies do not have established organizational practices to help them understand how to translate their existing products and services into those supported by technology—that also produce revenue. And these existing companies are challenged by new start-ups who begin with the web and iPhone as their platform of reference, rather than the store or book.

As technology evolves across all these industries we need to acknowledge that the people who make the products are less and less like the people who use them. We cannot rely on the intuition of the builders guided by their inner sense of what works to produce things that work for people. Their personal experience is not reflective of the experience of the real users and the customers. This is the same challenge that software companies faced years ago when CD was developed. The users were not the developers and the customers took no pleasure in figuring out how the software worked. We moved from creating tools for developers to editors for regular people. We moved from computers that calculated, presented forms, and offered command line interfaces to WYSIWYG drag and drop, and the web. Product development came to be about developing for the average everyday worker and the person at home, at school, and at play. This challenge simply continues in industry after industry as more technology is integrated into that industry. The product managers, business managers, analysts, user experience experts, designers, coders, IT administrators, and all others who pride themselves on understanding and using technology are not the users, but they are the builders. And so, as organizations, we need to bridge the gap between the users of the products and the makers of the products. Thus, the CD was designed. CD is a process for understanding people and driving that understanding into the design of products and systems.

43.2 CONTEXTUAL DESIGN OVERVIEW

CD is a step-by-step process for helping a team collect field data and use it for the purpose of defining and designing products, systems, websites, mobile devices, consumer electronics, and so on. It takes a team from understanding the customer through analysis of that data to build a coherent view of the practices and experiences of the customer/user. Using that consolidated data, CD helps the team to interact with the data to produce a high-level product or solution concept. This constitutes the first phase of CD: requirements and high-level solutions. This first phase of CD is useful for any kind of requirements and concept generation where understanding the day-to-day behavior of people is necessary. So, for example, medical device companies may use only this phase of CD because innovations and solutions are a matter of science, not software. Or, when a company wants to characterize a market to see if it is a reasonable area to expand into, the team may use only this first phase of CD. A "no" answer in this case is as valuable as a "yes"—it saves money and ends arguments.

If a product that can be materialized and prototyped as the solution (such as a software, or a hardware device), the second phase of CD is appropriate. These steps help a team to work out the details of the design and validate the product/solution direction while gathering low-level requirements and finalizing the user interaction design. Validation of the high-level solution concept is critical to ensuring success; iterative design has long been recognized[4] as critical for determining the detailed design. Because the initial concept is generated from understanding the user—rather than just collecting a list of desired features—it is more likely to be the kind of product or system that resonates as valuable with people. But in all product development the devil is in the details—so this second phase of CD works out those lower level details in a collaborative process with the user.

CD is a front-end design process that can fit with any corporate methodology. Today, many user experience teams are challenged to work within the agile development process. CD for Agile is a variant of CD that works with agile methods. (See Section 43.4.5.) But CD can also be an integral part of process reengineering, providing methods to validate process storyboards with stakeholders and users. It can support internal development where phase 1 might be done within the business but phase 2 within IT. Of course, CD works with more traditional development processes as well.

One cornerstone of CD is that it acts as a framework or scaffolding for putting structure into a part of the engineering process that is typically unclear: how to get to the most important requirements to drive the next version of a product or system. It helps teams step through an organized process to collect data and generate solutions, all the while involving stakeholders and team members. Section 43.3 is a description of key techniques and some issues.

43.3 PHASE 1: REQUIREMENTS AND HIGH-LEVEL CONCEPT

43.3.1 Contextual Interview—Getting the Right Data

In order to design a product that meets users' real needs, designers must understand the users and their practices.* But as we have discussed, designers are not usually familiar with or experienced in the user activities they are supporting. If they operate from their gut feelings, they rely on their own experiences as users. Generally, designers are more tolerant of technology than average users, so they are not representative of end users.

On the other hand, requirements gathering is not simply a matter of gathering requirements by asking people what they need in a system, like gathering pebbles from a beach. A product is always a part of a larger practice. It is used in the context of other tools and manual processes. Product design is fundamentally about the redesign of work or life practice, given technological possibility. In addition, practice cannot be designed well if it is not understood in detail.

You cannot simply ask people for design requirements, in part because they do not understand what technology is capable of, but more because they are not aware of what they really do. Because the everyday things people do become habitual and unconscious, people are usually unable to articulate their practices. People are conscious of general directions, such as identifying critical problems, and they can say what makes them angry at the systems they use. However, they cannot provide day-to-day details about what they are doing. This low-level detail of everyday practice is critical to ground designers in what is needed before they invent how technology might augment the process.

The challenge of getting this design data is designing a technique to get at the data that is unconscious and tacit. The first step of CD is Contextual Inquiry, our field data gathering technique that allows designers to go out into the field and talk with people about their work or lives while they are observing them. If designers watch people while they work, the people do not have to articulate their practices. If they do blow-by-blow retrospective accounts of things that happened in the recent past, people can stick with the details of cases using artifacts and reenactments to remind them of what happened. Contextual Inquiry overcomes the difficulties of discovering tacit information.

In CD, the cross-functional design team conducts one-on-one field interviews with users in their workplaces (or life spaces), focusing on the aspects of the practice that matter for the project scope. The Contextual Interview lasts about 2 hours and is based on the following four principles that guide how to run the interview:

- Context—While people work, gather data in the workplace and focus on the activities they are doing.
- Partnership—Collaborate with users to understand their work; let them lead the interview by doing their work. Do not come with planned questions.
- Interpretation—Determine the meaning of the user's words and actions together by sharing your interpretations and letting them tune your meaning. When immersed in their real lives and real works, people will not let you misconstrue their lives.
- Focus—Steer the conversation to meaningful topics by paying attention to what falls within the project scope and ignoring things that are outside of it. Let users know the focus so they can steer, too.

The Contextual Interview starts like a conventional interview, but after a brief overview of the practice, it transitions to ongoing observation and discussion with the user about that part of the practice that is relevant to the design focus. The interviewer watches the user for overt actions, verbal clues, and body language. By sharing surprises and understandings with users in the moment, users and designers can enter into a conversation about what is happening and why it is happening, and the implications for any supporting system. As much as possible, the interviewer keeps the user grounded in current activity, but can also use artifacts to trigger memories of recent activities.

The fundamental intent of the Contextual Interview is to help designers get design data: low level, detailed data about

* Some products, systems, and websites support the way people work, keep businesses running, or help users find needed information. Other products, systems, and websites address games, other entertainment, or consumer information to support life decisions. To gather data for these consumer products we have to look at people's life practice. To simplify language, this chapter will use practice to mean both work and life practice.

the structure of the practice and the use of technology within that practice. Contextual Inquiry, which is based on observing people in the context of their practices while they do their normal activities, has become standard in the industry as the best way to get this necessary design data.

43.3.1.1 Challenge: What Is the Field?

Collecting data in the field allows us to see the whole context of the users' practice: where they do their activities, how they do them, the tools they use, people they interact with, interruptions, breaks, and organizational props. Being in their own environment also ensures that the work we observe and discuss is real to them. It is not an artificial task in which they have little motivation. Being in the field also ensures that we can see them using not only the target product but related products; not only the target activities but related activities. As a result, field data lets us see adjacencies and constraints as well as the details of the actual practice. Adjacencies are key when looking for places for product expansion.

Today we know that a conference room, a usability lab, or a coffee shop is *not* the field—unless they happen to be where the real work is done. These locations engender out-of-context discussion and opinion ungrounded in real experience; or they impose artificial tasks that do not have the real load, context, or motivation of real life. Clearly a structured questionnaire or a conventional interview (directed by the interviewer) is not the same as following the work as it unfolds with the user. Only in that context will we see unarticulated and tacit experience.

But budgets are tight and today we have remote meeting software. What about a virtual field visit—is not that still the field? I heard of one company that conducts such "visits" with all the developers sitting behind the one person conducting the "field interview"—just like they used to sit behind the one-way mirror of a usability lab. We know that developers resist traveling, and that it is expensive. But a group experience like this does not engender the intimacy needed for genuine human connection and valid data between the interviewer and the interviewee. And this setting feels much more like the one-way mirror of the usability lab or focus group. Users know they are being watched—they feel that they are on stage.

Even if we maintain the one-on-one interviewing relationship, we lose the larger physical, social, and work context. Yet "traveling overseas" without the cost of travel is seductive. So this is our rule of thumb: *if* the scope of the project is small so that all the work/activity is performed online and in a tool, and *if* you have already done most of your field interviews in the field (60%), then adding on a few remote interviews to extend your reach can make sense. In the end, engineering is always a trade-off. We think this is the best way to make this one.

43.3.1.2 Choosing Customers Depends on Project Scope

The number of people that should be interviewed is directly related to the project scope. The wider the scope, the more people need to be interviewed to cover that scope. Our rule of thumb for a small scope, such as top 10 problems, usability improvement, next product release, or checking a planned design, is six to ten users from three to five businesses covering one or two roles. The more roles you need to cover and the more contexts (type of business, characteristic of person, and geographic location), the more people you need to interview.

When analyzing the project, first identify the job titles or roles targeted and the people that support the activities to be studied. Remember, the work or life practice that users engage in is what counts—if one person is called a system administrator and the other a database administrator, but they do the same work as it relates to the project, it does not matter that their job titles are different. We interview a minimum of three individuals per role; four is better.

If the product is to support people across different contexts (e.g., different industries) what matters is whether the contexts imply a different practice. In real estate, for example, the work is structured differently within the industry: a group of small, distributed agencies, a large corporate real estate company, and an in-house real estate representative. In each situation, the communication, sharing, and work management is likely to differ, creating three different practice patterns. To cover real estate, you need to collect data in all three contexts. We try to interview at least two businesses or independent set of people per practice pattern.

If, for your industries, there are no changes to practice patterns, then simply touch multiple industries without worrying about overlap. In this case, wide diversity is best.

The goal in selecting users to interview is to get enough repetition in the practice so that each role and contextual variable has three or four interviews that represent it, remembering that any one person may represent several of the contextual variables. As long as you have overlap, you will be able to find the common structure and key variations in the practice. Remember, paper prototype interviews later will expand the number of contexts and roles represented in the whole project.

43.3.2 Interpretation Sessions—Creating a Shared Understanding

Contextual Interviews produce large amounts of customer data, all of which must be shared among the core design team and with the larger, cross-functional team of user interface (UI) designers, engineers, documentation people, internal business users, and marketers. Traditional methods of sharing through presentations, reports, by e-mail do not allow people to truly process the information or bring their perspectives into a shared understanding. CD overcomes this by involving the team in interactive sessions to review, analyze, and manipulate the customer data. We recommend a cross-functional team of two to six individuals composed of, for example, a user researcher or business analyst, an interaction designer, a product manager, a developer, and so on. At least two team members should be full time on the project—but others can participate less often.

The interpretation session is a 2-hour commitment which should occur within 48 hours of the field interviews. The team gathers in the design room, where the interviewer tells the story of the interview from handwritten notes and memory. Plan to have at least three individuals in an interpretation session for each interview.

Team members ask questions about the interview, drawing out the details of this retrospective account. One person is the recorder, typing notes online. The team member who did the interview retells the story being captured. Any additional participants listen and point out the key issues to be captured. Each team member brings a different perspective to the data, whereas open discussion enables the team to arrive at a shared understanding. Participants ask questions, triggering the interviewer's memory and eliciting more data than would be available from a designer working alone. When the discussion sparks design ideas, they are captured in the notes.

The notes are displayed on a monitor or computer projector so that everyone can see them. These notes capture the key practice issues, cultural observations, breakdowns, successes, task patterns, design ideas, and any other interpretation or issue that has relevance to the project scope. Later these notes are transferred to sticky notes and used to build the affinity diagram.

While we audio record interviews for backup; we do not transcribe the recordings or do videotape analysis. Both transcription and video analysis take too long, and video analysis limits the perspective to one person. Since videotape and photos are so easy with new technology, teams may choose to capture specific images and a recording for a later highlight tape to be shared with key stakeholders to give them a flavor of the users. But we believe that the return on investment (ROI) on tape analysis is simply not worth the time and effort—and slows down the process. Interpreting within 48 hours produces the detail that is needed by a design team and creates another context for sharing the experiences. You can include one additional stakeholder in each interpretation session as a way to share data as it unfolds.

43.3.3 Work Modeling—Revealing the Structure of the Practice

Work models capture the practice of people in diagrams. Work models reveal the structure of the practice from different points of view. Each point of view helps the team understand what is going on in the world of the user from that aspect of the practice. Practice taken in as a whole can be overwhelming; each work model abstracts out a different aspect of the practice to reveal what people do and care about from that point of view only. The team can discuss and invent for that part of the practice without getting overwhelmed. Different work models are important for different kinds of projects; part of project planning is to pick the correct models to use.

Work models are captured by modelers during the interpretation session. Relevant data is recorded in each diagram as it is revealed during the retelling of the interviewing story. Models are in addition to capturing the key issues, as described above.

- The *sequence model* (see Figure 43.1) is equivalent to a task analysis. It shows each step required to perform a task in order. A sequence model represents the activities of someone who will use the system. The sequence model also shows the breakdowns in the practice (as do all the models). The consolidated sequence models become the basis for redesigning the steps of the activities. This is the most basic work model and is used in nearly every project.
- The *flow model* (see Figures 43.2 and 43.3) depicts people's responsibilities and the communication and coordination required to support the work. The flow model reveals the actual human process used by a workgroup or individuals within an organization irrespective of time or order. When consolidated, the flow model is the key model for finding the core workgroups to support and for redesigning processes—both formal and informal. The flow model is important any time when a product or system is trying to support a formal or informal business process like selling, bill paying, buying, or collaborating as a team.
- The *cultural model* (see Figure 43.4) reveals the influences on a person, a group, or an organization, whether external to the company (such as law) or internal company policies (standards). It reveals the cultural milieu in which the product will have to succeed. When consolidated the cultural model is the key model for identifying the value proposition for the system and revealing influences on buying behavior. The cultural model is important any time when a project is focused on decision making such as buying, or is concerned about cultural values conflicting across populations.
- The *physical model* (see Figure 43.5) shows the physical layout of the work or home environment and the constraints it imposes on the design. The physical model captures the footsteps between places, the role of distance, and the use of space. It shows the way people physically structure their work environment and work space. The physical model captures the structure and flow of work as it is manifest in space. When consolidated, the physical model can reveal both how to redesign work within the space and also how to support work online that was previously manual. The physical model is always helpful—along with pictures—for helping the team get a flavor for the physical environment of the user. But it is particularly important for the design of kiosks or other technology that will reside in space.
- The *artifact model* (see Figure 43.6) shows how artifacts are structured and used during the performance of tasks. The artifact model is a key when a team is trying to take a paper artifact, like a medical

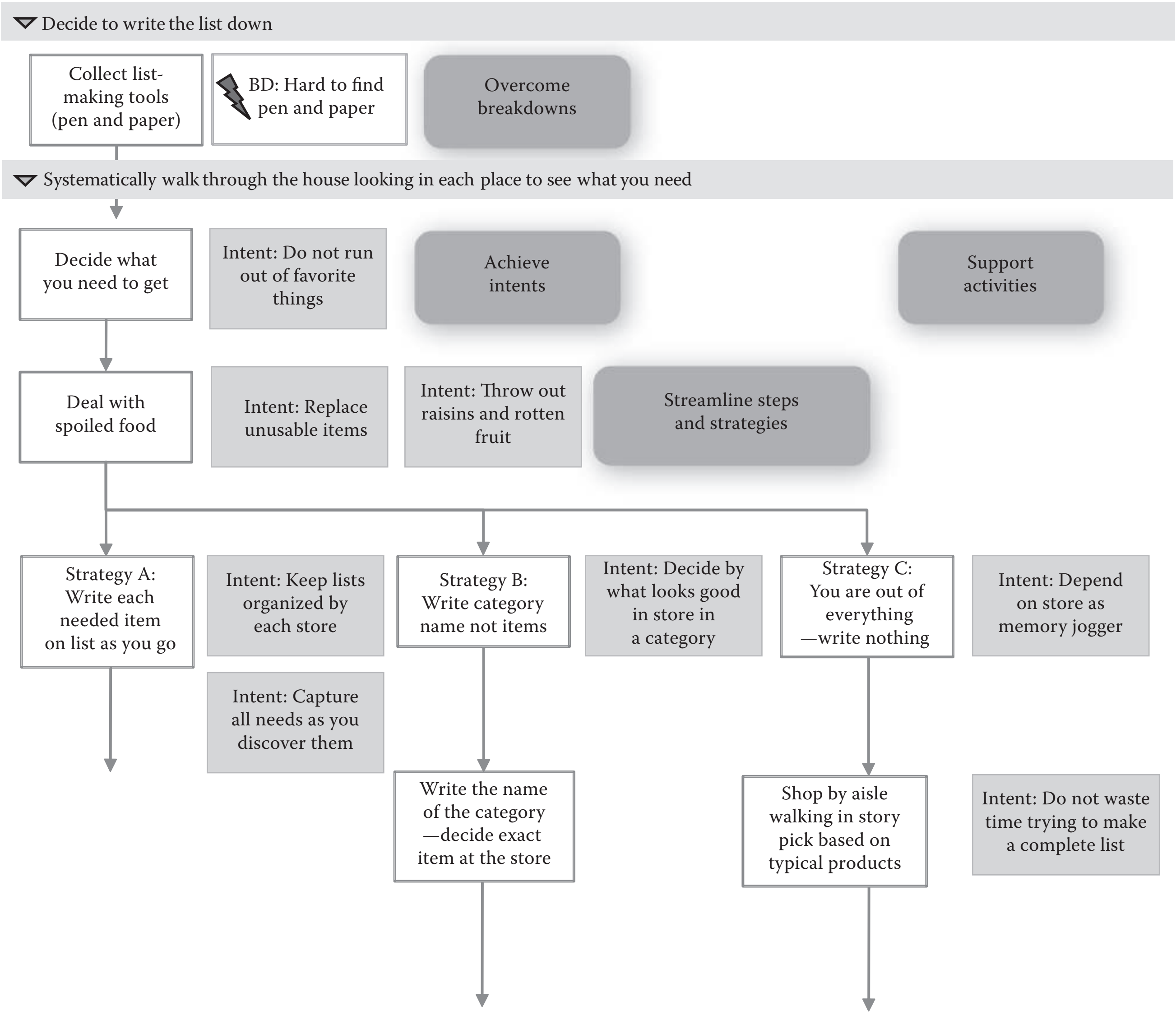

Decide to write the list down
Collect list-making tools (pen and paper)
BD: Hard to find pen and paper
Overcome breakdowns
Systematically walk through the house looking in each place to see what you need
Decide what you need to get
Intent: Do not run out of favorite things
Achieve intents
Support activities
Deal with spoiled food
Intent: Replace unusable items
Intent: Throw out raisins and rotten fruit
Streamline steps and strategies
Strategy A: Write each needed item on list as you go
Intent: Keep lists organized by each store
Strategy B: Write category name not items
Intent: Decide by what looks good in store in a category
Strategy C: You are out of everything —write nothing
Intent: Depend on store as memory jogger
Intent: Capture all needs as you discover them
Write the name of the category —decide exact item at the store
Shop by aisle walking in story pick based on typical products
Intent: Do not waste time trying to make a complete list

FIGURE 43.1 A sequence model showing the steps and strategies for a task.

record, and put it online. Analysis of existing artifacts identifies their intent, usage, structure, and information. Artifact models merge with physical models for designs for automobiles or technologies in the home. Space and usage of physical things reveal what is happening with the use of a complex product in space. The consolidated artifact model brings all the variations across users into one model so that they can be considered together for the redesign.

43.3.3.1 Challenge: Working as a Distributed Team

More and more companies are depending on distributed teams. Distributed teams have the advantage that data can be collected from different locations with less travel, which saves time and cost. However, any distributed team that has to be more organized is dependent on the latest collaborative technology, and needs to work harder to develop a shared understanding.

The interpretation sessions are easy to do in a distributed fashion. If you have collaboration or virtual meeting software, you can use that to run a distributed meeting. Simply display your word processor in a collaboration space so all can see the notes as they are typed. Get everyone on the phone and start the session as usual. As the interviewer tells the story, the recorder captures the issues and others can call out design ideas and ask questions to get at the details of what happened.

If you are capturing sequence models, capture those online in a word processor, switching the shared display from the interpretation notes to the sequence model as needed. Capture other models on paper, and periodically stop to check for correctness using video. If that is not possible, have them reviewed later by the other team.

The key to success is keeping everyone engaged. Be sure that everyone has a role and is actively participating. If someone has been quiet for a period, check in with him or her. Consider having both sides of the phone responsible for some type of data capture to ensure involvement.

Each bubble in the flow model represents a role. A job title plays multiple roles. Behind each role is a list of responsibilities the role plays. This flow model shows a process with the key work groups coordinating at each step in the process. Both formal and informal activity is depicted.

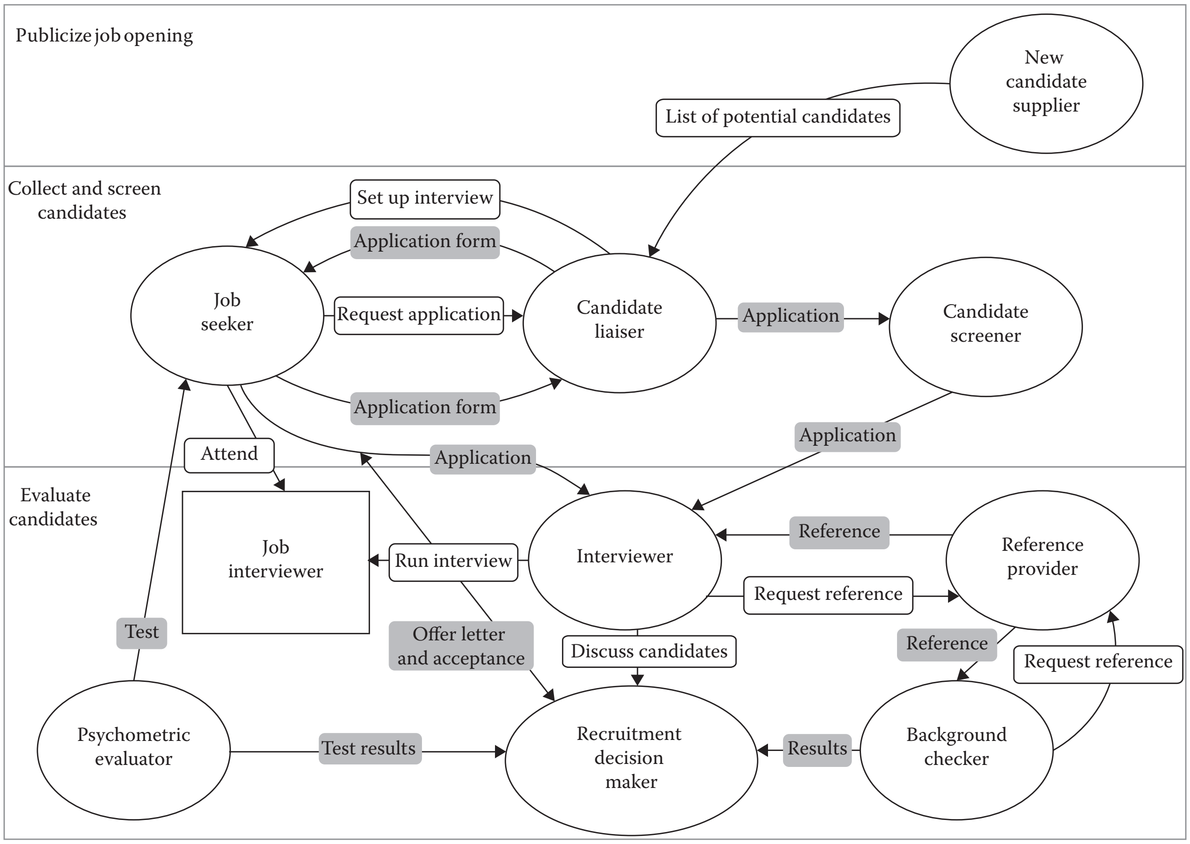


FIGURE 43.2 A flow model revealing the hiring process.

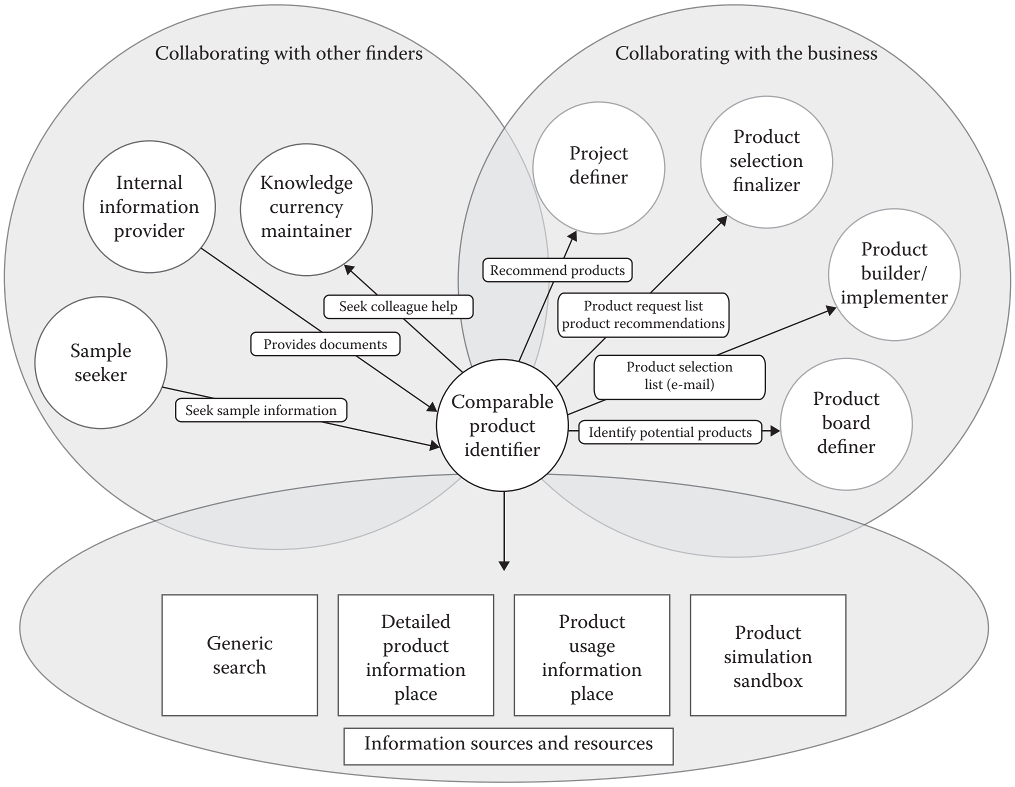

Collaborating with other finders
Collaborating with the business
Internal information provider
Knowledge currency maintainer
Project definer
Product selection finalizer
Product builder/ implementer
Recommend products
Seek colleague help
Product request list product recommendations
Provides documents
Sample seeker
Product selection list (e-mail)
Seek sample information
Comparable product identifier
Identify potential products
Product board definer
Generic search
Detailed product information place
Product usage information place
Product simulation sandbox
Information sources and resources

FIGURE 43.3 A flow model showing the process of finding on the web.

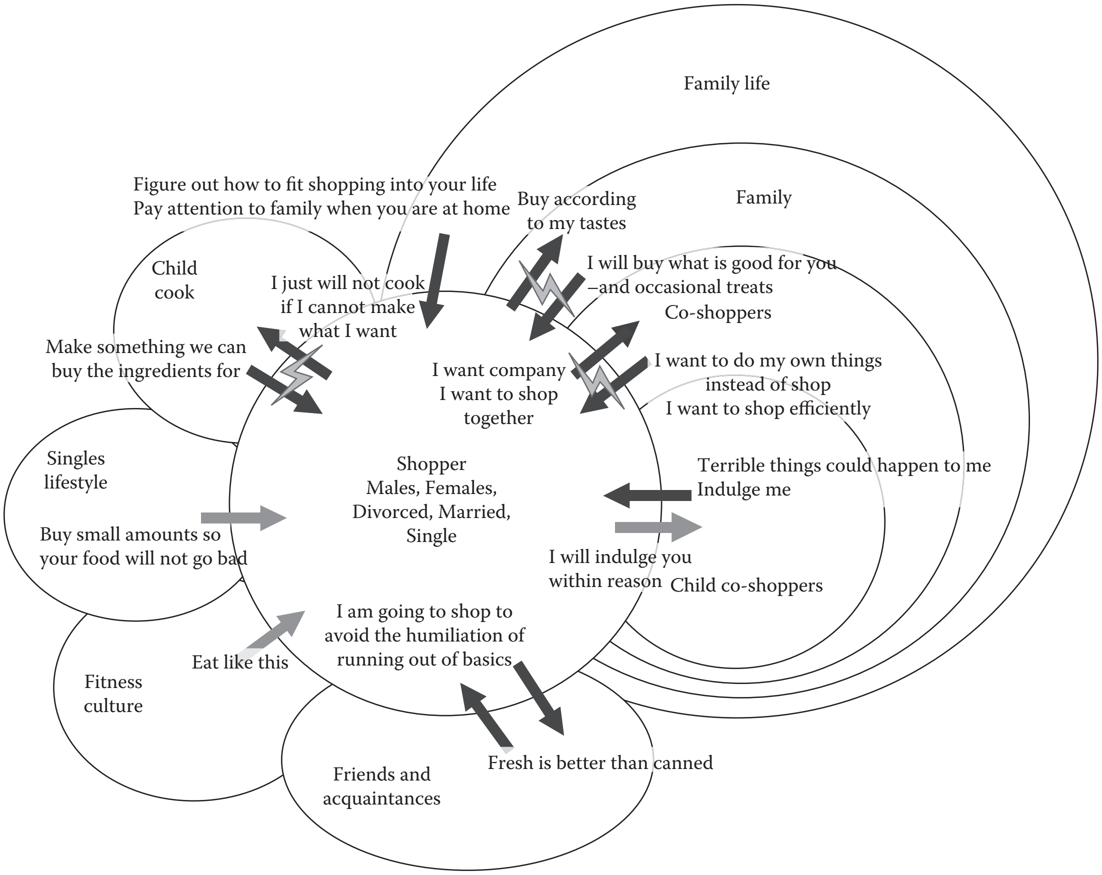

Family life
Figure out how to fit shopping into your life
Pay attention to family when you are at home
Buy according to my tastes
Family
Child cook
I just will not cook if I cannot make what I want
I will buy what is good for you –and occasional treats
Co-shoppers
Make something we can buy the ingredients for
I want company
I want to shop together
I want to do my own things instead of shop
I want to shop efficiently
Singles lifestyle
Shopper
Males, Females, Divorced, Married, Single
Terrible things could happen to me
Indulge me
Buy small amounts so your food will not go bad
I will indulge you within reason
Child co-shoppers
I am going to shop to avoid the humiliation of running out of basics
Eat like this
Fitness culture
Fresh is better than canned
Friends and acquaintances

FIGURE 43.4 A cultural model revealing the influences and pressures.

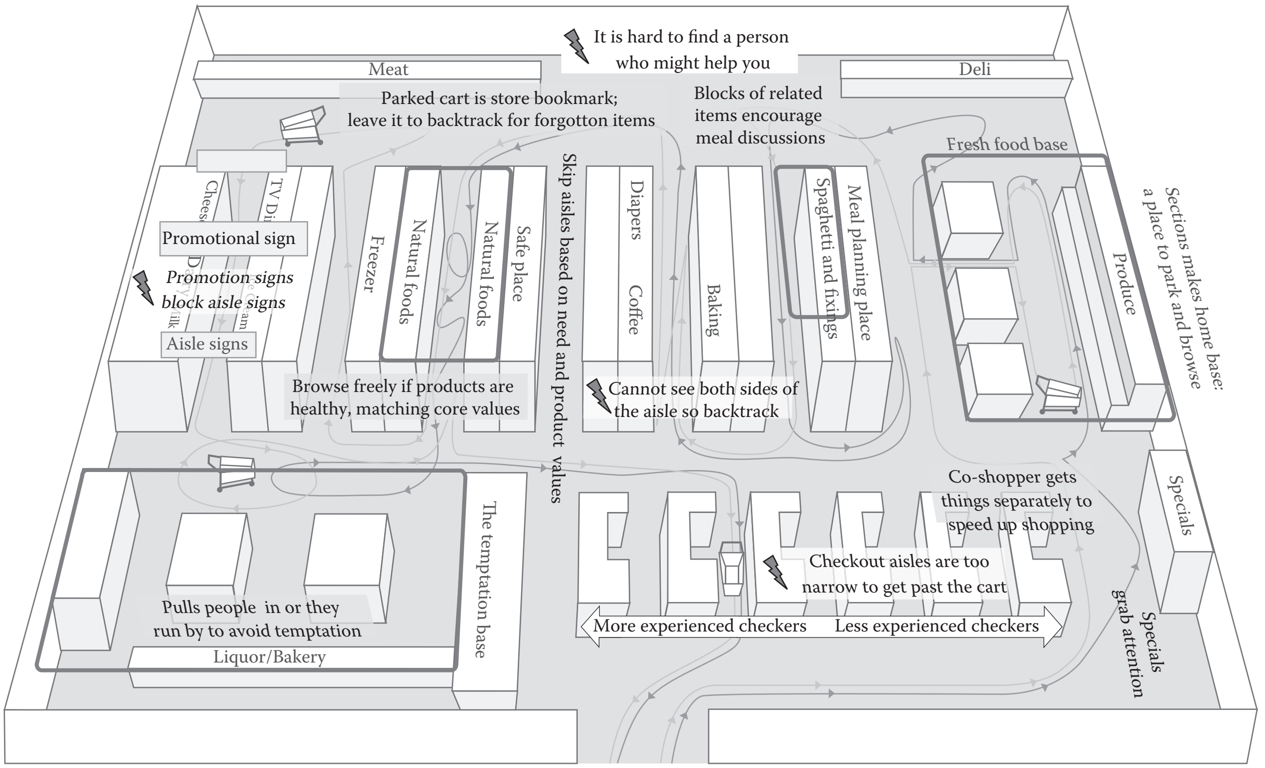

It is hard to find a person who might help you
Meat
Deli
Parked cart is store bookmark; leave it to backtrack for forgotton items
Blocks of related items encourage meal discussions
Fresh food base
Chees
TV Di
ilk
am
Promotional sign
Promotion signs block aisle signs
Aisle signs
Freezer
Natural foods
Natural foods
Safe place
Skip aisles based on need and product values
Diapers
Coffee
Baking
Spaghetti and fixings
Meal planning place
Produce
Sections makes home base: a place to park and browse
Browse freely if products are healthy, matching core values
Cannot see both sides of the aisle so backtrack
Co-shopper gets things separately to speed up shopping
Specials
The temptation base
Checkout aisles are too narrow to get past the cart
Pulls people in or they run by to avoid temptation
More experienced checkers
Less experienced checkers
Specials grab attention
Liquor/Bakery

FIGURE 43.5 A physical model showing the impact of the physical environment.

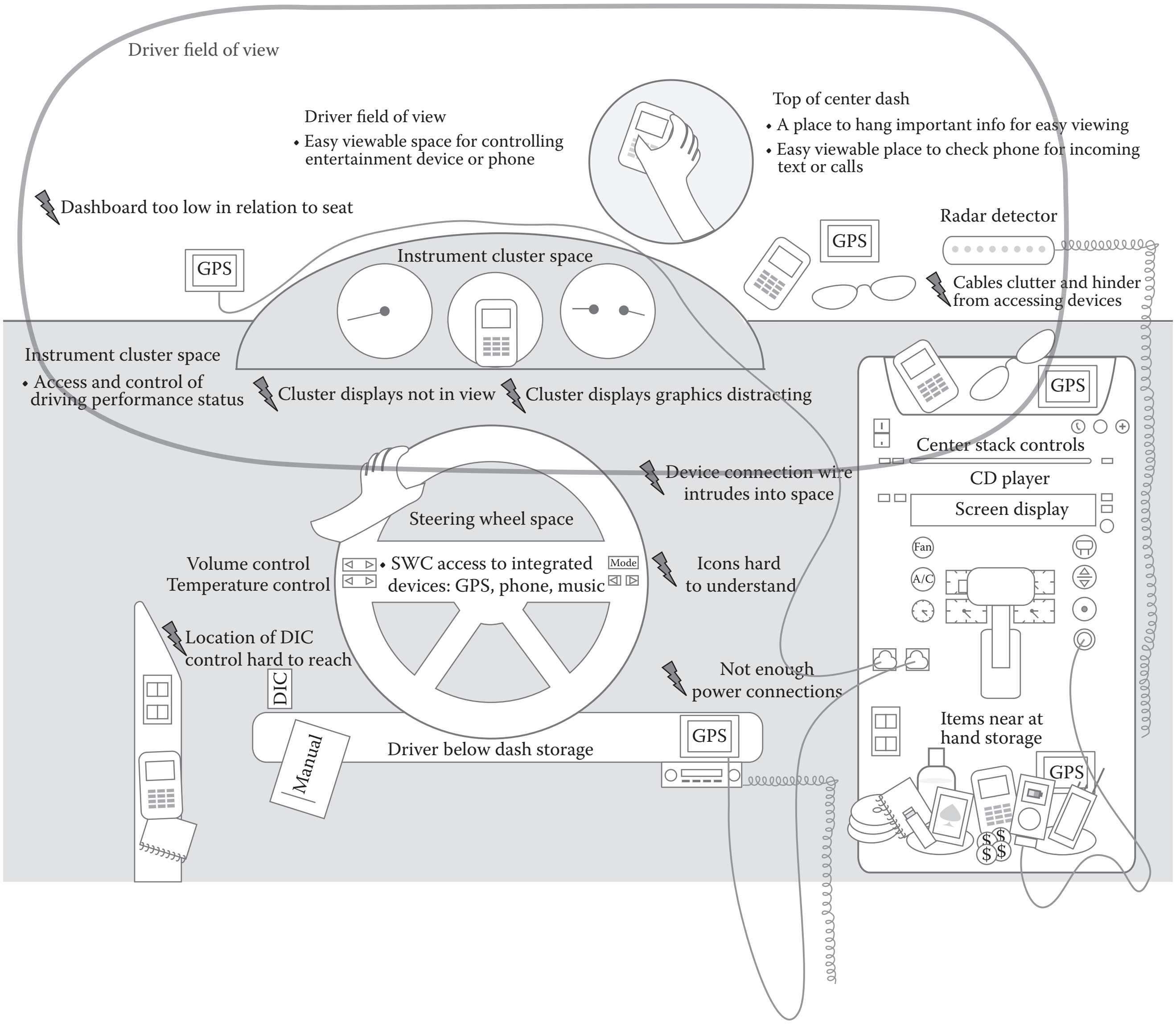

Driver field of view
Driver field of view
• Easy viewable space for controlling entertainment device or phone
Top of center dash
• A place to hang important info for easy viewing
• Easy viewable place to check phone for incoming text or calls
Dashboard too low in relation to seat
GPS
Instrument cluster space
GPS
Radar detector
Cables clutter and hinder from accessing devices
Instrument cluster space
• Access and control of driving performance status
Cluster displays not in view
Cluster displays graphics distracting
GPS
Center stack controls
Device connection wire intrudes into space
CD player
Screen display
Steering wheel space
Volume control
Temperature control
• SWC access to integrated devices: GPS, phone, music
Mode
Icons hard to understand
Fan
A/C
Location of DIC control hard to reach
DIC
Not enough power connections
Items near at hand storage
Manual
Driver below dash storage
GPS
GPS

FIGURE 43.6 An artifact model for the interior of a vehicle.

Finally, do not start distributed. Do your first interviews on the road together or in one location. Work out your process and your initial understanding of the users' activities, issues, and practices together. This becomes the bedrock of your shared understanding. After that, the distributed session works very well.

43.3.4 Consolidation—Creating One Picture of the Customer

Consolidating the notes and the work models creates a coherent representation of a market's or a user population's needs and activities. Then the design can address a whole population, not just one individual. The most fundamental goal of CD is to get the team to design from data instead of from the "I." If you walk in the hall and listen to designers talk, you will hear comments such as "I like this feature" or "I think the interface will work best this way." It is rare to hear, "Our user data says that the work is structured like this, so we need this function." It is natural for people who design to make a system hang together in a way that makes sense to them. But they are not the users, and increasingly, they are not in any way doing the activities of the people for whom they are designing. Getting designers, marketers, and business analysts out to the field and into interpretation sessions moves them away from design from personal preference. However, we do not want them to become attached to their users to the exclusion of the rest.

Product and system design must address a whole market or user population. It must take into consideration the issues of the population as a whole, the structure of work, and the variations natural to that work. The core intent of data consolidation is to find the issues and the activity structure and create a coherent way to see it and talk about it.

In CD, we consolidate the notes from the interpretation session into the affinity diagram and then consolidate each work model separately.

FIGURE 43.7 The affinity diagram.

The affinity diagram (see Figure 43.7) brings issues and insights across all users together into a wall-sized hierarchical diagram. The team prints the notes captured in the interpretation session onto sticky notes. The team then organizes all this data into groups, finding common underlying themes that cross the customer population. The process exposes and makes concrete common issues, distinctions, practice patterns, and needs without losing individual variation. Walking the affinity diagram allows designers to respond with design ideas that extend and transform the work. They write these on sticky notes and stick them right to the data that stimulated the idea. This encourages a culture of design from data over a focus on cool ideas generated from the "I."

The affinity is built from the bottom up, which allows the individual notes that come together to suggest groupings instead of trying to force them into predefined categories. Groups are labeled using the voice of the customer—saying what they do and how they think. A team of four can build an affinity diagram in 1–3 days. This is the place that we recommend using helpers—a one-day commitment can speed the process and expose stakeholders to the raw data building buy into the process and the data.

The consolidated work models show the detailed activity structure that the system will support or replace. For example, the sequence model shows each step, triggers for the steps, intents, different strategies for achieving each intent, and breakdowns in the ongoing work. Work practice redesign is ultimately about redesigning the steps in the sequences or the processes and practices represented by the different models. Whether the redesign eliminates or changes the steps or process, knowing the as-is process keeps the team honest, ensuring that any redesign does not forget the basic intents, collaborations, or activities that currently need to be performed. More often than not, technology introduces new problems into the practice by failing to consider the fundamental intents that people are trying to achieve. Redesign will better support the practice if it accounts for the existing practice—changing it consciously with consideration.

Every model helps the team think differently about the practice but the consolidated sequence is critical for nearly every system and acts as a guide for storyboarding or for streamlining any business processes. As such, it is nearly always used on a project. (See *Contextual Design* for guidance in consolidation.)

43.3.4.1 Personas Built with Contextual Data Reliably Focus the Solution

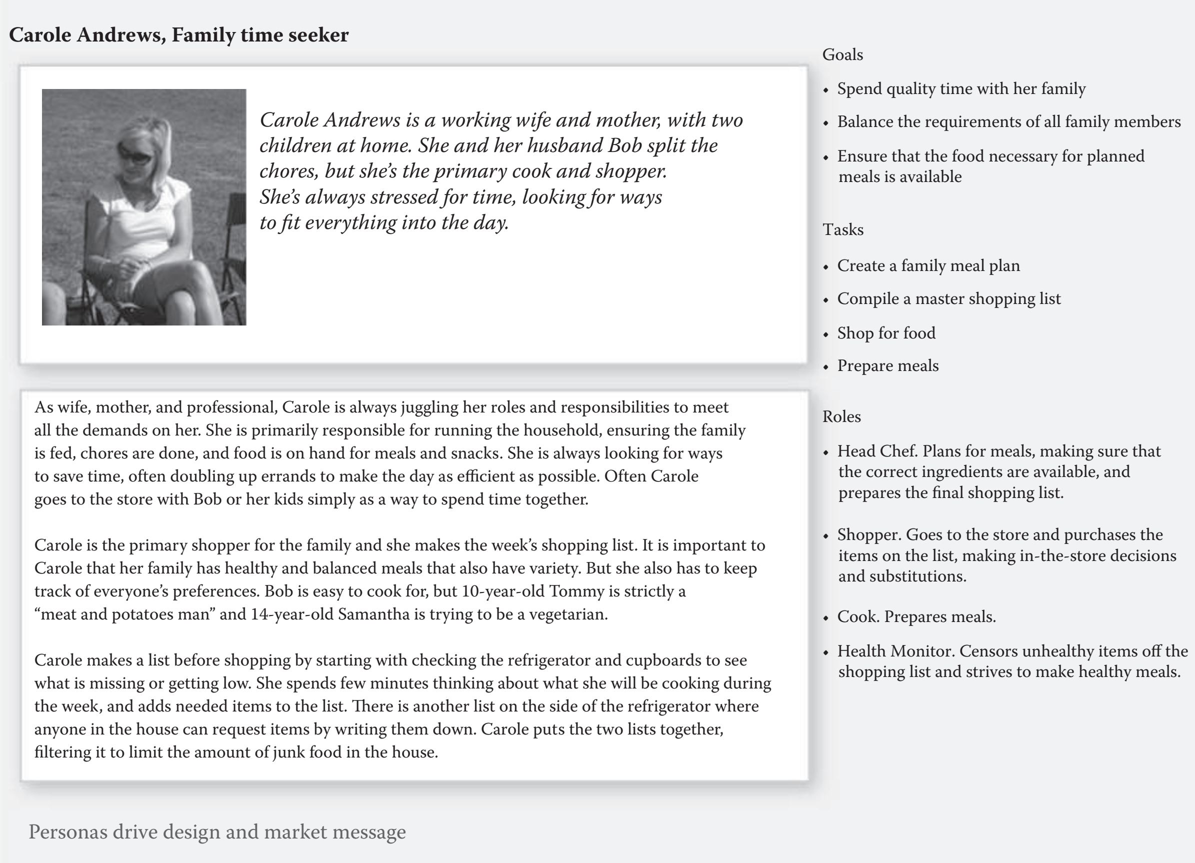
When your cross-functional team has worked together to gather, interpret, and consolidate customer data, you have developed a shared understanding of the users and their issues. You have not just built the data—you have experienced it in a way that others have not. Therefore, you may find the consolidated models and affinity to be evocative representations of the customer data you experienced, but others do not share that experience and cannot assimilate these models as easily.

Most teams need to communicate their understandings of user needs and their design plans to stakeholders—management, customer organizations, product groups, and so forth. These stakeholders, with no background in CD and no experience with the data, do not embody the memories of the real users from the field interviews. When they walk the data they can see the issues, needs, and breakdowns, but the power of knowing the users and their issues personally is not as poignant or personal.

Personas can help bring users alive and focus the stakeholders on the relevant issues, when they are built from rich contextual data. Popularized by Cooper,[5] a persona describes typical users of the proposed system as though they were real people. Their use is becoming more widespread, though with mixed success. According to Manning's research,[6] "A persona that is not backed by rich contextual data is not valid, which accounts for much of the mixed success." However, he goes on to say that, when backed by rich contextual data, they can help developers and designers not involved in the data collection focus on the needs and characteristics of their users. Anyone, trained in CD or not, can read this vignette and gain a sense of the typical user they are trying to support.

We build personas from the data collected and consolidated with CD to help focus on the characters we will vision about in the next step, to help stakeholders segment their market according to practice instead of typical demographics, to clarify branding and prioritization, and to bring the users and their needs to life for developers.

To build a persona, (see Figure 43.8) the team looks for core practice characters among the users, each characterizing a different way of doing the work. Expect anywhere from six to eight of these in a typical project. For each core character, find the base user: the user who most exemplifies this character. Then look at others who also manifest this character and borrow other relevant tasks, values, and life story elements to create your archetypical persona. This collection of exemplary facts then becomes the basis for your story of the named persona. With the rich user data in the affinity already grouped into issues, it is easy to harvest the data for the key elements that differentiate the different personae. Write a paragraph about the persona's



Carole Andrews, Family time seeker

Carole Andrews is a working wife and mother, with two children at home. She and her husband Bob split the chores, but she's the primary cook and shopper. She's always stressed for time, looking for ways to fit everything into the day.

As wife, mother, and professional, Carole is always juggling her roles and responsibilities to meet all the demands on her. She is primarily responsible for running the household, ensuring the family is fed, chores are done, and food is on hand for meals and snacks. She is always looking for ways to save time, often doubling up errands to make the day as efficient as possible. Often Carole goes to the store with Bob or her kids simply as a way to spend time together.

Carole is the primary shopper for the family and she makes the week's shopping list. It is important to Carole that her family has healthy and balanced meals that also have variety. But she also has to keep track of everyone's preferences. Bob is easy to cook for, but 10-year-old Tommy is strictly a "meat and potatoes man" and 14-year-old Samantha is trying to be a vegetarian.

Carole makes a list before shopping by starting with checking the refrigerator and cupboards to see what is missing or getting low. She spends few minutes thinking about what she will be cooking during the week, and adds needed items to the list. There is another list on the side of the refrigerator where anyone in the house can request items by writing them down. Carole puts the two lists together, filtering it to limit the amount of junk food in the house.

Goals

- Spend quality time with her family
- Balance the requirements of all family members
- Ensure that the food necessary for planned meals is available

Tasks

- Create a family meal plan
- Compile a master shopping list
- Shop for food
- Prepare meals

Roles

- Head Chef. Plans for meals, making sure that the correct ingredients are available, and prepares the final shopping list.
- Shopper. Goes to the store and purchases the items on the list, making in-the-store decisions and substitutions.
- Cook. Prepares meals.
- Health Monitor. Censors unhealthy items off the shopping list and strives to make healthy meals.

Personas drive design and market message

FIGURE 43.8 A persona built from Contextual Design data.

project-related life; get a representative picture; list the typical tasks, roles, goals, and other practice characteristics relevant to your project focus, and you have built a persona based on reliable customer data that you know represents your user population.

43.3.5 Visioning a New Work Practice

Visioning is about invention. However, design of technology is first design of the story showing how manual practices, human interactions, and other tools come together with your product or system to better support the whole practice. Visioning is the CD technique to help teams tell that story. Visioning is a vehicle to identify needed function in the context of the larger practice. Visioning ensures that teams put off lower-level decisions about implementation, platform, and UI until they have a clear picture of how their solution will fit into the whole of the practice. Teams commonly focus too much on low-level details instead of the full human-technical system. This is one reason systems often break the way users perform activities and fail to deliver something the market wants. Therefore, the primary intent of visioning is to redesign the practice, not to design a UI. Because a visioning session is a group activity, it fosters a shared understanding among team members and helps them use their different points of view to push creativity.

In CD, the core team visions a solution, but this is another time for bringing in stakeholders and helpers. A good visioning session has four to six participants. If helpers have been involved in the interpretation session and affinity building, they may want to participate in the visioning session. Again, this is about a 2-day commitment, which allows a wider team to participate in creating the solution direction.

The first step to a visioning session is to walk each consolidated work model in turn, immersing the team in customer data so their inventions will be grounded in the users' work. During the walk, team members compare ideas and begin to get a shared idea of how to respond to the data. Our rule is that no one gets to vision unless he or she has participated in walking the data. Without this rule, the process is no longer data-driven; anyone can walk in and offer their pet design ideas. Simply walking the customer data naturally selects and tailors preexisting ideas to fit the needs of the population. Since we will evaluate the visions based in part on fit to the data, knowing the data is important.

During the visioning session, the team will pick a starting point and build a story of the new practice. One person is assigned to be the pen. That person draws the story on a flip chart, fitting ideas called out by the team members into the story as it unfolds. The story describes the new practice, showing people, roles, systems, and anything else the vision requires. The team does not worry about practicality at this point; all ideas are included. Creating several visions allows the team to consider alternative solutions.

After a set of visions is created, the team evaluates each vision in turn, listing both the positive and negative points of the vision from the point of view of customer value, engineering effort, technical possibility, and corporate value. The negative points are not thrown out but used to stimulate creative design ideas to overcome objections. When complete, the best parts of each vision and the solutions to objections are brought together into one, named as synthesized practice redesign solution (see Figure 43.9).

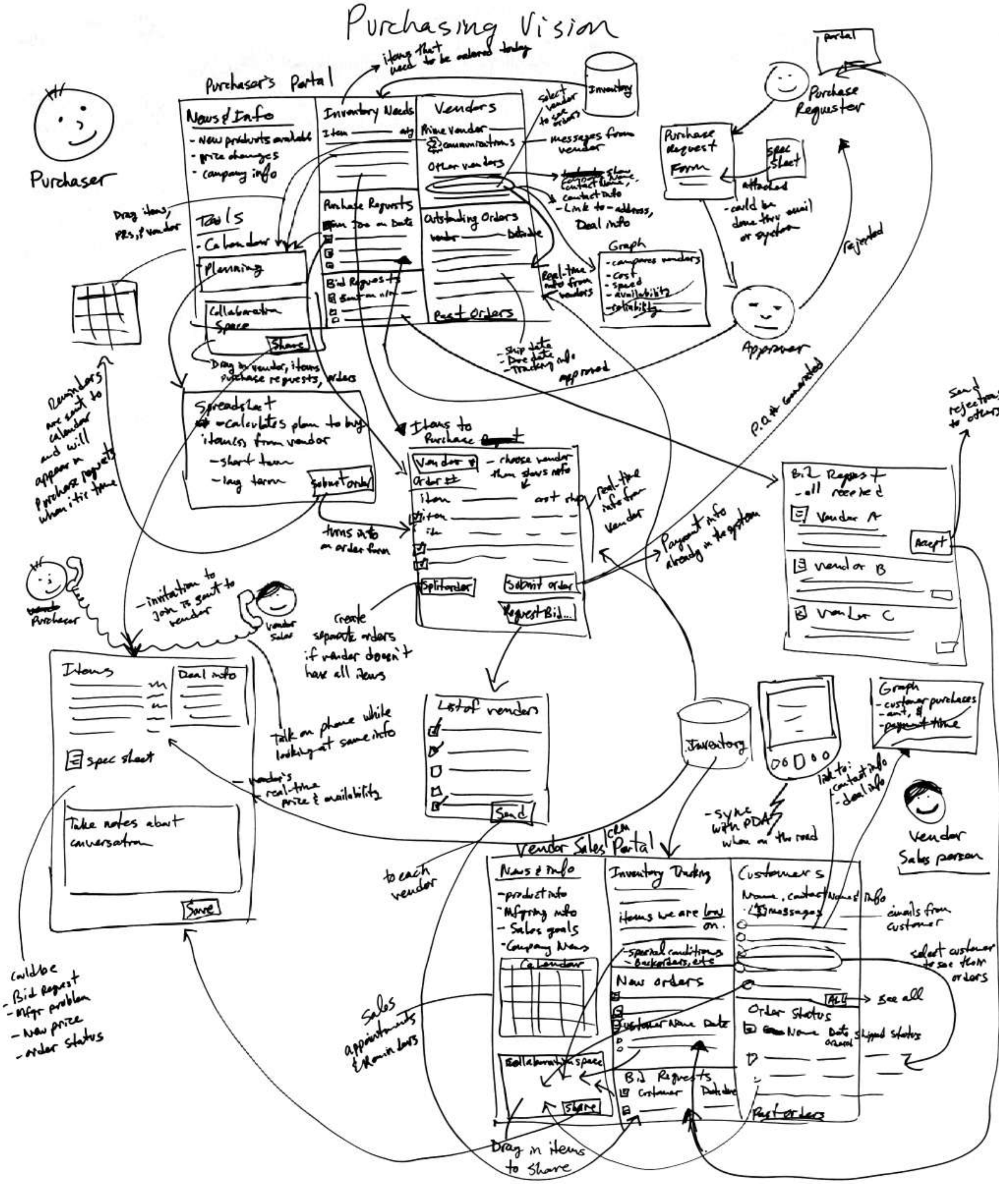


FIGURE 43.9 A complete, synthesized vision.

43.3.5.1 Challenge: You Cannot Invent from Customer Data

Every team we work with raises this claim. Customer data tells you what it is, not what it could be. How can you see the future by looking at the past? Here is what we say.

Every invention supports a real need—otherwise, why would anyone want it? Invention is a response to some life or work practice by a designer or technologist who, on seeing a need and knowing the technology, imagines a new possibility. Edison did not invent the idea of light; he saw candles and gas and electricity and invented light bulbs. Bricklin did not invent spreadsheets; he saw how paper spreadsheets are used and knew what technology could do.[7] The developers of WordPerfect worked in the basement below a secretarial pool. Nobody invents entirely new things that fulfill no need and contribute in no way to human practice; they invent new ways to fill existing needs and overcome existing limitations. As people incorporate these products and new ways of working into their lives, they reinvent it by adopting and adapting the new way of working. If designers are out there, seeing people living their lives with, without, and in spite of technology, they can see future directions for technology.

A vision is only as good as the team's combined skill. Customer data is the context that stimulates the direction of invention. However, no invention is possible without understanding the materials of invention: technology, design, and practice patterns. The visioning team needs to include people who understand the possibilities and constraints of the technology. If the team is supposed to design web pages and none of them has ever designed a web page, they will not be able to use web technology to design. When the people who always designed for mainframes were told to design windowing interfaces, they replicated the mainframe interface in windows. This is why we recommend that design teams include people with diverse backgrounds representing all the materials of design and the different functions of the organization. Only then will an innovative design, right for the business to ship, emerge.

We do not ask customers what to make; we understand what they do. Customers are not aware of the details of their practices, do not know the latest technologies, and do not know what your business is capable of—so they cannot tell you what to invent. We do not ask them. Instead, we understand what they are doing and capture it systematically. We immerse a design team—who does understand technology, the practice, and the business—in that data and let them vision. However, we are not done. The vision has to be right for people, so we take it out and test it. We let people test-drive the future in our paper mockups and let their tacit knowledge of their lives shape and direct our vision.

43.3.6 Scoping the Project—The Half-Way Point in CD

The vision, produced by a cross-functional team immersed in customer data, is the jumping off point for concurrent design:

- Interaction design: The vision guides the detailed design to produce the overall user experience architecture and final interaction design.
- Engineering: The vision contains implementation assumptions and challenges that must be looked at for viability before the company can commit to the vision.
- Marketing: The vision is the story of the new practice—the basis for communication to customers. Sales and marketing are always based on storytelling: how the new product will benefit the customer. Sharing the story in user groups, focus groups, and individual conversations is a good way to gauge sales point (the excitement that the new product will generate).
- Business planning: The vision can be used to drive marketing surveys and investigations to flesh out the business case.
- Business (or enterprise) process design: The data itself reveals the root causes of problems and suggests which processes to study more and measure. It also provides a representation of the as-is process to drive process mapping and as-is use for case development. The vision represents the new human-technical system that could be put in place to streamline the process. Rather than starting by improving the processes and then designing the system to support it, the vision redesigns the process and the technology as one integrated whole.
- Testing: The consolidated models drive test case development.
- Documentation: The vision communicates what the product or system is so they can start the introduction to the user's manual.

Because the vision is the center of the design, revealing direction for a system or product based on real user data, some companies may wish to transition to their existing processes after this step. But whatever the team does at this point they must pick the part of the vision that they will carry forward into detailed design and validation. A vision produces more than can be developed by any team in a reasonable amount of time. Many teams take 3–5 years to fully develop the ideas produced in their vision. In an agile or rapid CD project, the scope is constrained in the beginning—but even then a vision may produce design direction for several releases.

43.4 PHASE 2: DETAILED DESIGN

43.4.1 Storyboarding—Working Out the Details

Too often when people design, they break the existing practice because they jump from their big ideas to low-level UI and implementation design. As soon as designers start focusing on technology, technology and its problems become their central design concerns. How technology supports the practice is subordinated.

The steps and strategies of a practice, being tacit, are easy to overlook. If we are out in the field watching them, we can see them. This is what is captured in the sequence and other models. Storyboarding keeps the team honest and the design clean. Guided by the affinity diagram and consolidated models, the vision is made real in storyboards. Storyboards ensure that the team does not overlook any intents and steps that are critical to the practice. Even if the practice is changed, we have to think through the details of how it will be changed to ensure that adoption is easy.

Storyboards are like freeze-frame movies of the new practice (see Figure 43.10). Like storyboarding in film, the team draws step-by-step pictures of how people will perform activities in the new world of the vision. Storyboards include manual steps, rough UI components, system activity and automation, and even documentation use.

Because they focus on practice redesign, storyboards prevent the design team from prematurely delving into too much detail. They are guided by customer data, and after each task has been thought through and sketched, the team reviews it to ensure that it remains true to the customer data. This does not mean that no invention happens, but the team must account for the steps and other data elements of the affinity. They must look at them and make a conscious decision about how to handle them. They might change all the steps and even eliminate whole sequences; as long as people can still achieve their fundamental intents, the change will work. When teams forget or ignore the user's intent, the design is in trouble.

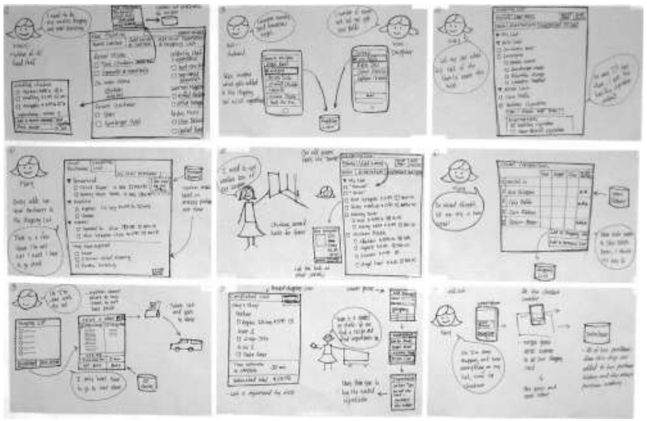


FIGURE 43.10 Storyboards are like freeze-frame movies.

43.4.2 User Environment Design

A good product, system, mobile device, or web page must have the appropriate function and structure to support a natural workflow within it. System design really has three layers. At the top, the visual design presents and provides access to the function, structure, and workflow provided by the system. At the bottom, the implementation makes that function, structure, and flow happen. But the core of a product is the layer in between: the design of the behavior of the system to support the user's redesigned work. Just as architects draw floor plans to see the structure and flow of a house, designers need to see the floor plan of their new systems.

Hidden within the storyboards are the implications for the system floor plan—the User Environment Design (UED) (see Figure 43.11). The UED formalism represents a set of focus areas or places in the system that provides support for coherent activities. A place might be a window, web page, dialog box, or pane. The UED shows each part of the system—how it supports the customer's work, exactly what function is available in that part, and how the customer gets to and from other parts of the system—without tying this structure to any particular UI or implementation design. The function defined in the UED drives functional specification and

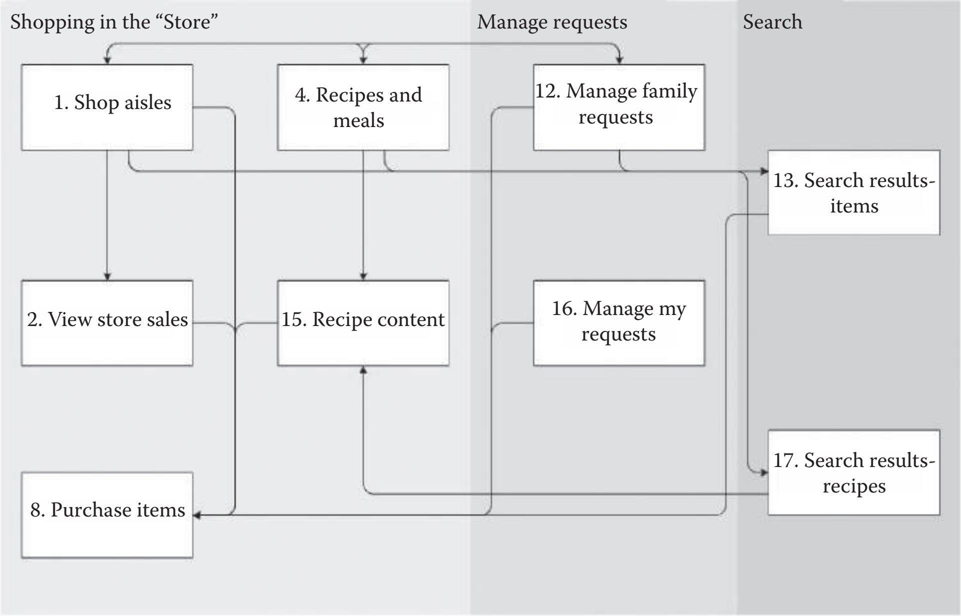


FIGURE 43.11 The User Environment Design.

implementation level use cases. The function in each focus area becomes the specification for the part of the UI that will support that function.

In creating a UED, the team walks the storyboards and derives the implications of what the system needs to provide. As the implications of storyboard after storyboard are rolled into the UED, the team starts to see the best way to structure the system. This system structure now represents the system that, if built, will actualize the vision as it has been worked out in the storyboards. Whether this system will be valued by the users has yet to be tested. Any design team can anticipate needed function only up to a point. Therefore, after working out the preliminary UED, the team mocks up each focus area in paper and tests it in mock-up interviews. Through iteration with the users, the UED stabilizes, the UI paradigm stabilizes, and the lower-level requirements are solidified.

43.4.3 User Interface Design and Mockup

After visioning, the team will work out their first cut at how to present their ideas in a UI. The team defines how to lay out the needed function and information represented in the UED into a set of rough screens representing the core places in the system. These places in the UI may augment an existing interface, add new places and features implied by the vision, or represent the changes that were worked out in more detail within the storyboards.

However, this function and initial layout is yet untested with users. To ensure that you have the right solution, testing and iteration is essential. To test your design concepts and clarify your functions, we recommend constructing a paper prototype and testing it with your user population in paper prototype field interviews. Users do not understand models or even storyboards when presented with them. However, users can talk UI talk, or form factor talk for physical products. So to test ideas, we move quickly through initial storyboarding and system structure to a rough UI to get back to the users with our ideas represented as UIs.

Many people have talked about paper prototype testing and its value, including Snyder.[8] Build the paper prototype using normal stationery supplies (see Figure 43.12). Card stock provides a stable background to simulate the screen. Post-its effectively simulate anything that might be moved during an interview, such as pull-down menus or buttons. Sample content should be put on a removable sheet so that users can replace it with their own real content during the

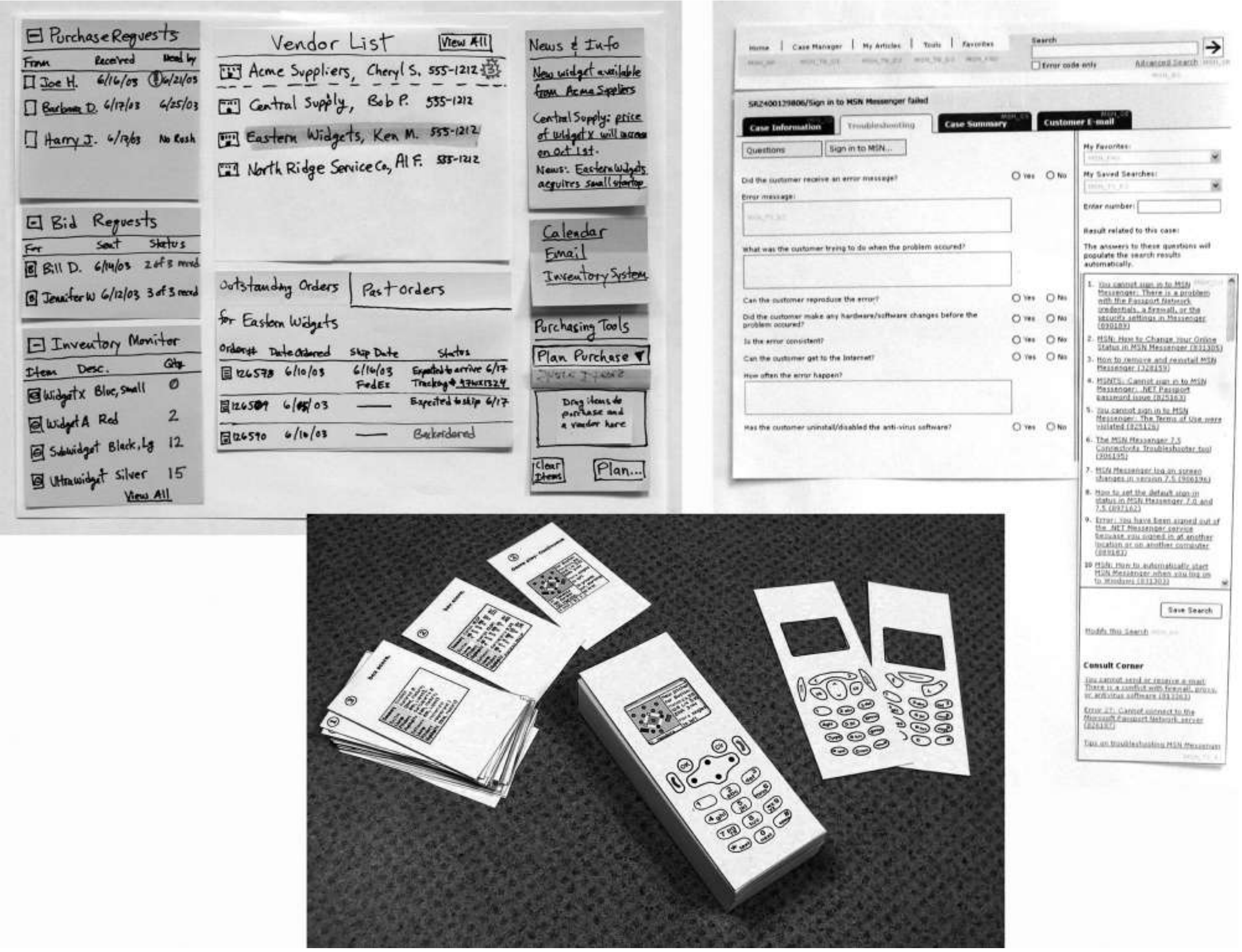

Purchase Requests
From Received Need by
Joe H. 6/16/03 6/21/03
Barbara D. 6/17/03 6/25/03
Harry J. 6/17/03 No Rush
Bid Requests
For Sent Status
Bill D. 6/14/03 2 of 3 recvd
Jennifer W 6/12/03 3 of 3 recvd
Inventory Monitor
Item Desc. Qty
Widget X Blue, Small 0
Widget A Red 2
Subwidget Black, Lg 12
Ultrawidget Silver 15
View All
Vendor List
View All
Acme Suppliers, Cheryl S. 555-1212
Central Supply, Bob P. 555-1212
Eastern Widgets, Ken M. 555-1212
North Ridge Service Co, Al F. 555-1212
Outstanding Orders
Past orders
for Eastern Widgets
Order# Date Ordered Ship Date Status
126578 6/10/03 6/16/03 FedEx Expected to arrive 6/17 Tracking#
126509 6/05/03 Expected to ship 6/17
126590 6/10/03 Backordered
News & Info
New widget available from Acme Suppliers
Central Supply: price of widget X will increase on Oct 1st.
News: Eastern Widgets acquires small startup
Calendar
Email
Inventory System
Purchasing Tools
Plan Purchase
Drag items to purchase and a vendor here
Clear Items
Plan...
Home | Case Manager | My Articles | Tools | Favorites
Search
Error code only
Advanced Search
SR2400129806/Sign in to MSN Messenger failed
Case Information
Troubleshooting
Case Summary
Customer E-mail
Questions
Sign in to MSN...
Did the customer receive an error message?
Error message:
What was the customer trying to do when the problem occured?
Can the customer reproduce the error?
Did the customer make any hardware/software changes before the problem occured?
Is the error consistent?
Can the customer get to the Internet?
How often the error happen?
Has the customer uninstall/disabled the anti-virus software?
○ Yes ○ No
My Favorites:
My Saved Searches:
Enter number:
Result related to this case:
The answers to these questions will populate the search results automatically.
Save Search
Modify this Search
Consult Corner

FIGURE 43.12 Examples of paper prototypes.

interview. Web content should be laid out, for example, products or content types to provide a context for discussing the structure of the content and its layout. For hardware designs, use other kinds of props to simulate the hardware. The final prototype is rough and hand drawn, but it represents both the system's structure and its behavior.

Mock-up interviews help designers understand why design elements work or fail and help to identify new function. These interviews are based on the principles of Contextual Inquiry given earlier. We test the paper prototype with users in their contexts to keep them grounded in their real practices. Users interact with the prototype by writing in their own content and manipulating and modifying the prototype. The partnership is one of co-design. As the user works with the prototype, following a task they need to do or did in the recent past, the user and designer uncover problems and adjust the prototype to fix them. Together the user and interviewer interpret what is going on in the usage and come up with alternative designs.

After the design has been tested with four to six users (depending on the scope of the project), we redesign to reflect the feedback. Multiple rounds of interviews and iterations allow testing in increasing levels of detail, first addressing structural issues, then UI theme and layout issues, and finally detailed user interaction issues.

We find that three rounds of testing result in a product or system that is valued by the users and has finalized the structure of the product, the user interaction design, the content type and tone, any navigation structure or information architecture, and clarified business rules. However, if you are moving into an agile process and additional testing will occur within the sprints, two rounds may be sufficient. You may also choose to put the third round online and use remote interviews to test the overall look in a more realistic environment. After two rounds of face-to-face testing in paper, a third round of remote online interviews can work well.

43.4.3.1 Challenge: Paper Versus Online Testing

Designers like tools. We are so often asked why we use paper mockups instead of online prototypes. We create mockups in paper because they are fast to assemble and they allow the user and interviewer to make changes to the interface in the moment. Hand-drawn paper prototypes make it clear to the user that icons, layout, and other interface details are not central to the purpose of the interviews; it keeps the user focused on testing structure and function. The rougher the mockup, the easier it is to test structure and function.

As the interface is tested, it starts to stabilize. At round three of the mock-up interviews, we begin to solidify the interaction design and build our mockups in wire frames. However, we still print and cut the wire frames up to allow for real co-design with the user and quick changes in the moment.

Toward the end of the prototype rounds, we use running prototypes to test low-level usability function that simply cannot be tested on paper.

How are mock-up interviews different from demos, rapid prototyping, or usability tests? The primary issue in answering this question is, where did the design being tested come from? Any time you iterate an existing design, you can expect not more than a 15%–20% change to the structure that is being presented. People will have an opinion about what you are showing them; it is much harder to find out that what you are showing them is not fundamentally valued. The question is what kind of data you can get in each context:

- Demonstrations at a user conference or in a focus group assume that people are aware of their practices, that out of the context of real use they can predict and project what they would need or want, and that their knowledge of their activities is not tacit but known such that they can bring it to bear in a discussion. We do not find this to be true. Demos and focus group discussions are better tests of sales point than function. They measure excitement with the product and clarity of the sales story.
- Rapid prototyping allows users' tacit knowledge to shape and tune a system that is being presented. Since the context of the discussion is to try to tune and shape what is already designed, the fundamental structure of the system and its value is rarely challenged. Some designers or technologists decided that the presented function was needed and should be structured in a particular way. CD does rapid prototyping, but only after systematically using customer data to produce the basic design to be tested. In this way, customer data shapes the whole of the design, not just the final testing.
- Usability testing can be another kind of rapid prototyping and as such suffers from the same challenges if the tested system is not originally designed from customer data. This is why usability professionals are at the forefront of pushing processes like CD into their organizations. They know that usability has to be built into the product during definition phase, not left to the end of the process. At some point, paper mock-up interviews are no longer good enough to test fine interaction with a tool. It may be possible to take a running prototype to the user on a laptop. If not, testing in the usability lab is an alternative. Even when testing a running prototype we recommend using Contextual Inquiry techniques: re-creating the user's own work tasks, discussing issues as they occur, and proposing design solutions.

43.4.4 Integrating Contextual Techniques into Standard Processes

Some companies have software methodologies within which any user-centered design process must fit. In general, CD can fit into any software or system development methodology. Most methodologies define a series of stages, each with deliverables and milestones. Few define specific ways of gathering requirements, so CD can easily be fit within the requirements step of these methodologies. Methodologies differ in the deliverables they require for their milestones. However, the natural data and design artifacts of CD can be used easily as inputs into any methodology's format.

Corporate methodologies typically follow a set of steps such as

- Business case. Define the marketing or work improvement rationale for building a new or revised product or system. Use phase 1 of CD to characterize an external market and generate product concepts that can then be further investigated and validated with focus groups, surveys, and other market intelligence. Or use phase 1 to understand the current business work practice—including the informal steps that are documented nowhere—and design business process and support system improvements. Use the vision with stakeholders to show the business value of the new direction.
- Requirements gathering. Gather user needs and overall business needs to guide building the product or system. Phase 1 of CD will also get the detailed data on issues and tasks needed to develop a product/system and phase 2 will result in system requirements at the level of the UI and core function. Some companies would be satisfied with requirements extracted from untested storyboards. Paper prototyping tests the requirements as well as the user interaction, so we recommend that you finalize your requirements after you define and test your UI.
- Design. Detailed design of the product or system, both the UI and underlying technology, making sure it can be implemented. Phase 2 of CD also results in the high-level design of the final layout and structure of the system and the function as it is presented to the users. Following multiple rounds of iteration and redesign you will need to scope the actual deliverable. Then specification and final design can be completed for the shipped product/system.
- Implementation. Code and test the UI and the overall system. Here you test the running system first with running prototypes and then during Alpha and Beta field tests using Contextual Inquiry to gather low-level and tactical changes.

All organizations have some sort of method or process that user-centered design must fit into. The challenge is how to communicate to the organization in a language they understand, and how to ensure enough time is budgeted for field research. Once you understand the methodology of your company, different steps of CD may occur at different points—but if you do them all (or a reasonable variant) you can ensure that your product/system will work for people.

Any methodology can use the consolidated customer data to drive brainstorming of recommendations, or the visioning process to generate a more systemic response to the data. Visioning is close enough to brainstorming, a known technique in any company, that it should be able to be easily adopted.

Similarly, paper prototyping is widely accepted as a means of testing design ideas and refining them. We recommend doing it in the field, with real user cases and not the canned test scenarios of some usability testing. This kind of prototyping is great for getting feedback from users and working out the details of a particular design. Sometimes this is the best first step in getting user data into a design and letting developers and designers alike see how their systems are being received.

CD can also become the basis for the development of personae to augment the design process. Usability testing is built into the process such that good user experience is a natural outcome of any CD project. If you just want to get started infusing user data into their processes, you can just do Contextual Inquiry interviews and build an affinity. In only a few weeks, you can begin to understand your market or user population and generate solution concepts.

43.4.5 Challenge: Agile Development and UX

Agile development is a relatively new approach to producing software. In contrast to the traditional approaches that emphasize requirements analysis, design, and implementation as distinct phases, agile methods seek to minimize upfront planning in favor of producing working base levels quickly and often. Feedback from these base levels is used to ensure that the resulting product is useful. Scrum[9] and XP[10] (Extreme Programming) are two popular agile approaches.

Agile techniques were initially promoted by the development community to protect themselves from changing demands. From their point of view, requirements constantly change and are unreliable anyway—providing the system as defined by the requirements offers no confidence that the system will be acceptable to users. So huge amounts of effort are poured into creating very large requirements documents which are unreliable and which change over time anyway. Then years go into developing the system—and by the time the system is delivered, requirements have already changed and the system is obsolete.

Accordingly, agile methods favor face-to-face interactions over formal documentation. Rather than writing the requirements down, agile teams seek to have the customer or product owner tell them directly about a needed function, right before coding on that function starts. To organize development, functions are written as "user stories"—simple index cards that describe one requirement at a high level. All the details of the design of that user story are worked out in conversation with the customer.

The customer role in an agile team is critical to its success. This role defines what needs to be done, prioritizes tasks, and works closely with the developers to work out each detail of the system. However, most agile teams are unable to put a real customer on the team—and when building for a user population or market, a single user's feedback is not enough. Consequently, they usually have a customer surrogate instead, who has all the difficulties we have discussed above in representing the real users. Scrum teams work with a product owner instead, who is given authority to make final decisions—but this product owner is not a real user either.

Agile methods organize development around short sprints, from 1 to 4 weeks in length. At the beginning of the sprint, the user stories to be implemented are selected. At the end, the resulting code is tested and the team reflects on their process. In theory, the project is re-evaluated at each sprint and could change direction completely. In practice, there is generally a strong expectation that the stories defined at the beginning of the project are the ones that will be implemented (just as they would have been in traditional development) and there is little time to rework stories that users aren't happy with.

Because agile teams distrust up-front design, they are impatient with user research. They would rather build something, test it, and change it instead of getting bogged down in a long design process. But it is very hard to develop a coherent UI within the constraints of short development sprints—and being developer-driven, agile methods do not provide for systemic design. Fortunately, agile teams in the industry have started to discover how difficult it is to produce a coherent product without doing some design work.

Agile development dovetails very nicely with user-centered design. Although there are some cultural clashes, the problems agile teams face are very much the ones user-centered design knows how to solve: how to have a reliable customer voice on the team; how to produce a systemic design to drive development; how to test and iterate that design with customers; how to do detailed UI design and testing in the middle of development sprints. We see CD integrated with agile as follows:

- Phase 0. We recommend starting an agile project with what many call a 'phase 0' or 'sprint 0'. This should last for 4 to 10 weeks, depending on the scope of the project and the number of roles to be supported. (Any single agile team can really only expect to implement for 2–3 roles, which limits the overall length of this phase.) At this time, the UX team can do Contextual Inquiry interviews with key roles, build an affinity and consolidate sequence models, vision with stakeholders, and do 0–2 rounds of paper prototype testing and iteration. The more rounds of testing they do the more stable the design; the fewer rounds they do the more they will need to test and iterate the low-level designs during sprints.
- Release planning. Development starts with a release planning session, in which the initial set of stories to be implemented is selected. The UX team participates in writing and prioritizing user stories. Each story describes an element of the vision developed in phase 0. The story does not have any of the design detail in it—"as a system manager I need to see what systems are down or offline at a glance"—might be the whole of a user story. These stories are prioritized into sprints, seeking to put the most important stories first and to produce a minimally useful system as quickly as possible.
- Sprint planning. Each sprint starts with a sprint planning session. User stories are selected for implementation in the next sprint, choosing only as many stories as the team thinks it can implement in the time. The team then plans tasks to complete the story. Some of these tasks should be for the UX designers—to work out the detailed UI design, consult with developers on system look and behavior, and test out any implementation. If multiple rounds of testing were done in phase 0, the story may need just low-level visual design and a sanity check with users; if few or no rounds were done, and then expect to need a few rounds of paper prototyping during the sprint or across multiple sprints to get the design right. Often, the UX work will take longer than the time of a sprint allows. In that case, expect to do the UX design work a sprint before actual implementation starts.
- Sprint execution. During the sprint itself, the UX designers perform their different roles. They go to users with prototype designs in paper or online. They communicate completed designs to developers who are starting work on stories. And they test completed code with users to ensure that the design as implemented meets users' needs.

43.5 CONCLUSION

The core of CD is getting customer data into the minds of product managers, designers, and developers. It provides the data that is needed to guide business decisions, prioritize requirements, identify how to streamline work, be clear on what will be of value to the user, and produce a high-quality user experience. Becoming user-centered is accessible today to any company that really wants to be user-centered. In these times when nontechnical products and services are becoming technical—when technology is becoming integrated into almost every aspect of life—design for people is even more important than it was 20 years ago. And as industries like publishing, banking, and retail are being challenged to create new ways to provide their services online, understanding the role of technology in people's lives and how to support them through products and services may be the difference between success and lost business.

REFERENCES

1. H. Beyer, and K. Holtzblatt. *Contextual Design: Defining Customer-Centered Systems*. San Francisco, CA: Morgan Kaufmann Publishers Inc., 1997.
2. K. Holtzblatt, J. Wendell, and S. Wood. *Rapid Contextual Design: A How-to Guide to Key Techniques for User-Centered Design*. San Francisco, CA: Morgan Kaufmann Publishers, 2005.

3. M. Gualtieri, *Best Practices in User Experience (UX) Design*. Forrester Research. Cambridge, MA, September 4, 2009.
4. Morgan, D. "Covert Agile—Development at the Speed of … Government?" in *Proceedings of the Agile 2009 Conference* (Agile 2009), pp. 79–83. Chicago, IL: IEEE Conference Publishing Services, 2009.
5. A. Cooper. *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity.* Indiana, IN: Sams Publishing, 1997.
6. H. Manning. *The Power of Design Personas.* Forrester Research. Cambridge, MA, 2003.
7. H. Beyer. "Calling down the lightning." *IEEE Software* 11, no. 5 (1994): 106.
8. C. Snyder. *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces.* San Francisco, CA: Morgan Kaufmann Publishers, 2003.
9. K. Schwaber, and M. Beedle. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Prentice Hall, 2001.
10. Beck, K. *eXtreme Programming Explained: Embrace Change*, second edition. Boston, MA: Addison-Wesley, 2004.

44 Grounded Theory Method in Human–Computer Interaction and Computer-Supported Cooperative Work

Michael J. Muller and Sandra Kogan

CONTENTS

44.1 INTRODUCTION

Grounded Theory Methods (GTM) are a set of practices for exploring a new domain, or a domain without an organizing theory (Glaser and Strauss 1967). The practices are strongly *grounded in the data* and the theory is said to *emerge from the data*. These practices provide intellectual rigor for organizing an inquiry (Corbin and Strauss 2008; Charmaz 2006a, 2009; Gasson 2004). Most GTM inquiries in human–computer

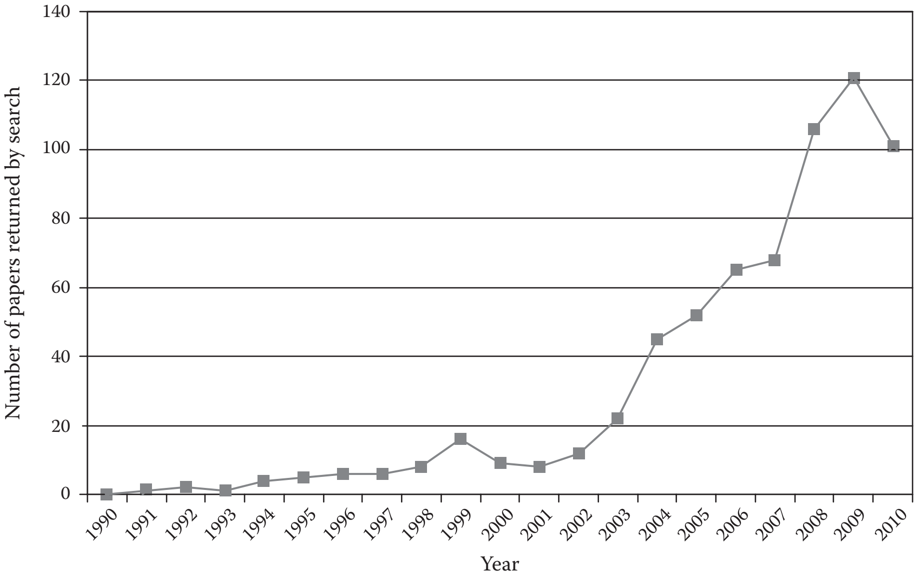


FIGURE 44.1 Grounded theory references in the Association for Computing Machinery Digital Library (2010 is an estimate based on data through May 2010).

interaction (HCI) and computer-supported cooperative work (CSCW) are based on qualitative data, but the same rigor can be applied to exploratory studies that use quantitative data, or a combination of qualitative and quantitative data (e.g., see contributions in Bryant and Charmaz [2007]; Morse et al. [2009]).

Grounded theory has been growing in importance. Hood (2007) reports figures from the Social Science Citation Index, in which grounded theory was mentioned in 101 journal articles during the 1970s, more than twice that number (296), in the 1980s, and six times that number (605) between 2000 and 2006. At the time of writing, the Association for Computing Machinery Digital Library lists 645 entries for the search “grounded theory,” with 180 articles on that topic during January 2009 to May 2010 (Figure 44.1).

44.1.1 Grounded Theory Addresses Human–Computer Interaction and Computer-Supported Cooperative Work Topics

In recent examples from the Association for Computing Machinery (ACM) Digital Library, grounded theory has been used in papers on software engineering, including developers’ orientation to new projects (Dagenais et al. 2010), metrics (Umarji and Seaman 2008), software testing and quality processes (Taipale and Smolander 2006), bug tracking (Bertram et al. 2010), and a developers’ open source community (Ge, Dong, and Huang 2006). Developers’ views on user interface design issues and team cohesion were also explored (Ferreira, Noble, and Biddle 2007; Whitworth and Biddle 2007), as well as their use of forecasting visualizations (Asimakopoulos, Fildes, and Dix 2009; see also Faisal et al. [2008] for end-user experiences with visualizations), along with developers’ social relations (Williams et al. 2007), their relationship to marketers (Jantunen and Smolander 2006), and the differences between their views and the views of their managers (Umarji and Seaman 2009). Managers’ views of software development were also explored (Adolph, Hall, and Kruchten 2008; Lee, DeLone, and Espinosa 2006; Morgan and Finnegan 2010).

In a more traditionally sociological framework, grounded theory has been used in the study of organizations (Sousa and Hendriks 2006), virtual teams (Sarker 2006; Sarker et al. 2001), and management of information systems methodologies (Goede and de Villiers 2003). In the organizational context, grounded theory has also been used to understand the transfer of training (Fincher and Teneberg 2007) and individual self-interruptions (Jin and Dabbish 2009).

Outside of the strict workplace context, grounded theory has helped to understand social phenomena in families (Dalsgaard et al. 2006), the needs of families managing collections of music (Sease and McDonald 2009), and viewing the contents of videos related to smartphones (see also Blythe and Cairns [2009]; Swallow, Blythe, and Wright [2005]). Accessibility needs have also been addressed (Kane et al. 2009; Savago and Blat 2009). In more personal domains, grounded theory has been used to study people’s formative experiences with computers (Schulte and Knobelsdorft 2007), impression management during disease (Mamykina et al. 2010), and value issues in games software (Barr et al. 2006).

Finally, in a more reflective way, grounded theory has been used to explore the needs of scholars in media studies (Kierkegaard and Borlund 2008) and for an examination of HCI methods (Nørgaard and Hornbæk 2006).

44.1.2 Grounded Theory Is a *Conceptual* Method

The “methods” in GTM are *not* a series of step-by-step procedures that are to be carried out with participants, or that are to be applied to data. Rather, the methods relate to *ways of thinking* about data (Glaser and Strauss 1967), and especially ways of making sense of the data (Charmaz 2006a,b), gradually and iteratively developing a theory to describe the data and the phenomena that are manifested through the data. Ideas, understandings, and new possibilities are recorded in a series of memos, which then form part of the structure of the report of the work. In the process of GTM, the theory is rigorously tested, again and again, by constant comparison with the data, and the series of tests both strengthens the theory and leads to new insights (Chiovitti and Piran 2003; Haig 2005; Stern 2007; Suddaby 2006). This combination of open-minded exploration and rigor is the hallmark of GTM (Bowen 2006; Stern 2007). It is also a perplexing paradox, and we will spend considerable effort in this chapter to make sense of that paradox.

44.1.2.1 Uses of Grounded Theory Method in Human–Computer Interaction and Computer-Supported Cooperative Work

Grounded theory originated in sociology (see Section 44.1.2.1.1), and has become a key methodology in certain fields such as nursing. In those fields, it is often used as the major organizing principle for an entire large project (see examples in Bryant and Charmaz [2006a, 2008]; Morse et al. [2009]). Within HCI and CSCW, GTM has also been used as a method for data analysis within a larger project, which is then integrated with more formal theories in the broader research literature. There are two major ways that researchers have applied GTM in HCI and CSCW.

44.1.2.1.1 Analyzing Data That Have Already Been Collected

Perhaps the most common usage in HCI and CSCW is to make sense of data that have already been collected. For example, a researcher might conduct a series of open-ended interviews with a predetermined sampling strategy. Once the interviews have been transcribed, the researcher could use GTM to discover the major themes that emerge from the interviews, and then develop a sense of conceptual categories among those themes, perhaps with high confidence about how several subthemes contribute to the same category. Then the researcher would set each category in relation to the other categories, finally producing a detailed analysis of the conceptual structure of how the informants described their experiences.

In this way, GTM is used to analyze data that have been collected in a relatively familiar HCI research program. In this structure of work, the domain is known; the type of data to be collected is known; the sampling strategy is known; and the number of informants has been predetermined. Most of the readers of this chapter have conducted research in this relatively familiar and straightforward manner.

44.1.2.1.2 Organizing an Exploratory Study

The second usage of GTM in HCI and CSCW is different, and is closer to the original intentions and claims of the methodology (Glaser and Strauss 1967). In this second usage, the researcher has only a general sense of the domain. The researcher collects an initial small sample of data, and then *immediately begins to construct a theory* of the domain. The researcher knows that the theory is based on insufficient data, and that therefore *the theory will be wrong in some respects*. The researcher analyzes his or her incomplete theory and determines its weakest point—a research strategy known as abduction (see also Peirce [1865/1982]; Reichertz [2007]). That weakest point becomes the focus for the next small sample of data. The researcher tests the theory with the intent of *learning how it fails* at its weakest point, and of making a stronger theory based on that failure.

The stronger theory is, of course, also incomplete, and shows the way to its own weakest attribute. Abductive testing continues with iterations of small data samples and further theorizing, that is, the researcher continues to try to make the theory fail. Each failure leads to more knowledge, and the theory becomes stronger, more detailed, and broader with each iteration. Finally, the researcher discovers that he or she is no longer learning anything new from each iteration, and the theory is ready to share with others.

This way of constructing theory follows a different form of scientific reasoning from the conventional paradigm of hypothesis testing (e.g., Popper 1968; Sanderson 2003; for discussion, see also Dodig-Crnkkovic [2002]; Greenberg and Thimbleby [1992]; Jaccard and Jacoby [2010]; Mackay and Fayard [1997]). Where hypothesis testing is often called *deductive* or *confirmatory*, GTM follows a form of *inductive* or *emergent* theory development (Glaser and Strauss 1967; Goulding 1998; Jaccard and Jacoby 2010). There is a large scientific and philosophical basis for inductive theorizing (e.g., Awbrey and Awbrey 1995; Dewey 1986; Eiter and Gottlob 1995; Menzies 1996; Patokorpi 2009; Peirce 1865/1982; Yu 1994), but it may be less familiar to many readers of this chapter. There is also a 400-year tradition of abductive reasoning, which is a core component—and a key differentiator—of grounded theory (Reichertz 2007).*

More controversially, there is a disagreement about whether grounded theory follows an objectivist research agenda (i.e., sensing the world *as it is*) or whether grounded theory is part of a post-objectivist, constructivist agenda that recognizes all scientific descriptions and theories as being contextualized in a particular culture, and potentially in the biography of a particular researcher (see Charmaz [2007, 2008]; more generally in HCI, see Kaptelinin et al. [2003]). The reader is entitled to know that our own views tend toward the constructivist position. However, most of the methodology of grounded theory can be applied across both sides of this disagreement, and we have been careful to present the material that is common to both perspectives in this chapter.

44.1.2.2 Strengths and Weaknesses of GTM: A Preview

Before we begin to provide details, it is fair to ask, *What is GTM good for in HCI and CSCW?* We provide some

* See also the International Research Group on Abductive Inference, http://user.uni-frankfurt.de/~wirth/.

quick answers here, and we will return to this question at the conclusion of the chapter:

- Grounded theory is useful to explore a new domain, or a domain without a dominant theory.
- Grounded theory is a useful method for *constructing* a theory of this new domain.
- Grounded theory is an excellent method for avoiding a premature conclusion about the domain. In medicine and forensics, we would say that GTM helps to avoid "confirmation bias," or the tendency to collect data that we expect to agree with our hunch (or our developing theory). The principle of abduction (Peirce 1865/1982; Reichertz 2007) is exactly opposed to confirmation bias, because it makes us avoid testing the theory at its strongest point, where we would expect it to succeed, and where the success will provide little new information. Rather, abduction teaches us to test the theory at its weakest point (Awbrey and Awbrey 1995), where we would expect it to fail, and where the failure will provide a large amount of information about how to improve the theory. Abduction helps us to be *surprised*, and surprise is often the gateway to new insights and the kind of "discovery" that is at the heart of the GTM (Reichertz 2007).
- Grounded theory is *not* a useful method for testing a hypothesis, or for trying to prove or disprove a theory (Suddaby 2006). The deductive, hypothesis testing methods are much better suited to that kind of problem.

We also think that GTM transforms a human weakness into strength. Most of us think about the data we are collecting. Most of us try to explain the data to ourselves and our colleagues, even while we are engaged in the early stages of a research project. In a conventional data collection discipline, this kind of informal theorizing is considered a problem. It can lead us to ask the wrong questions, or to bias the answers. It can cause us to make subtle changes in the way we analyze our data, or in the strategies we use for finding informants. We fear that we may unintentionally distort what was supposed to be a uniform process for collecting data in the same way from each informant. In conventional data collection, we try to avoid this kind of premature theorizing, and we try to discipline ourselves to prevent that theorizing from influencing how we collect data.

By contrast, GTM encourages us to theorize throughout the process of choosing where and how to sample, and throughout the period in which we collect and analyze our data. In GTM, we use our tendencies to think and to theorize as advantages. Rather than try to deny our thoughtfulness, we develop disciplines, such as abduction, to try to formalize our theorizing into a quality process that leads to new insights and new theories. As *thinking* scientists—as reasoners who are often passionate about our research—we can use the rigor of GTM to make us more fully human.

44.2 SYNOPSIS OF GROUNDED THEORY METHOD

Charmaz (2008) writes,

> Grounded theory methods consist of simultaneous data collection and analysis, with each informing and focusing the other throughout the research process. As grounded theorists, we begin our analysis early to help us focus further data collection. In turn, we use these focused data to refine our emerging analyses. Grounded theory entails developing increasingly abstract ideas about research participants' meanings, actions, and worlds and seeking specific data to fill out, refine, and check the emerging conceptual categories ...

Research using GTM usually begins with a broad, very shallow set of unorganized information (left side of the "Data" portion of Figure 44.2). Over time, through a series of disciplined procedures (Charmaz 2006a; Corbin and Strauss

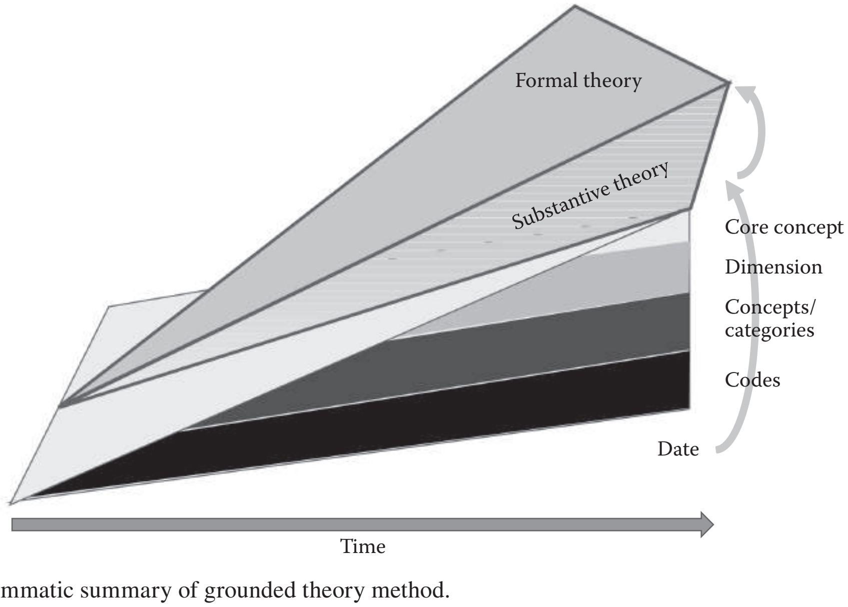


FIGURE 44.2 A diagrammatic summary of grounded theory method.

2008; Glaser 1978, 1992, 1998; Strauss 1987, 1993; Strauss and Corbin 1990, 1998), the information becomes more narrowly focused, and is understood in greater depth (i.e., moving from left to right in the "Data" portion of Figure 44.2) (Glaser and Strauss 1967). These disciplined procedures take the form of the following:

- *Coding* of the data in greater degrees of integration, breadth, and understanding (Star 2007), usually including
  - Open coding
  - Axial coding
  - Selective coding
- Development of *theory* through memo-writing (Charmaz 2006a,b; Dick 2005; Lempert 2007)
- *Theoretical sampling*
- *Constant comparison* of the developing theory with the data, always returning to the data (Glaser and Strauss 1967; Corbin and Strauss 2008)

Each of these concepts will be explained in the following sections.

As the data collection becomes more focused, the researcher begins to develop more and more powerful descriptions (theories) of the domain; in GTM, these are called "substantive theory" (Glaser and Strauss 1967) as shown in Figure 44.2. Finally, the substantive theory, which is developed from the data *internal* to the project, is brought into relationship with more formal theories from the research literature or from past investigations (Corbin and Strauss 2008; Lempert 2007). These more formal theories are *external* to the project, but are of course important in broader interpretations of the project, and in integrating the project's findings into the broader research literature.

44.3 BRIEF BUT NECESSARY HISTORY OF GTM

The history of Grounded Theory has been told many times (e.g., Bryant and Charmaz 2007; Charmaz 2006a,b, 2008; Locke 2001; Morse et al. 2009), and we will not rehearse it in detail here. However, it is helpful to know about a schism in the thinking of the two founders of the approach, and the "second generation" of grounded theorists (Morse et al. 2009) who have begun to heal that schism, and who are introducing new directions of their own (Figure 44.3).

Grounded theory was "discovered" during collaboration between Bernard Glaser and Anselm Strauss during the 1960s, and was first described as a methodology in their *Discovery of Grounded Theory* (1967). Glaser and Strauss had already worked together on topics of death and grieving (Glaser and Strauss 1965), and they continued their collaboration into the next decade (Glaser and Strauss 1968; Strauss and Glaser 1970).

In the logical positivist atmosphere of the 1960s in the United States, their insights were a revelation. Star (2007) described the impact of Glaser's and Strauss's early work as "a manifesto for freedom from the sterile methods that permeated social sciences at the time." Others were similarly moved; students gathered, and this new form of inquiry developed an enthusiastic and increasingly influential set of students, who in time went on to teach their own students about grounded theory.

44.3.1 Divergence

Subsequently, Glaser and Strauss proposed different types of procedures. Strauss favored a formalized set of methods, published in a series of theoretical works (Strauss 1987, 1993). With Juliet Corbin, Strauss provided and repeatedly revised a set of core procedural descriptions of methods (most recently

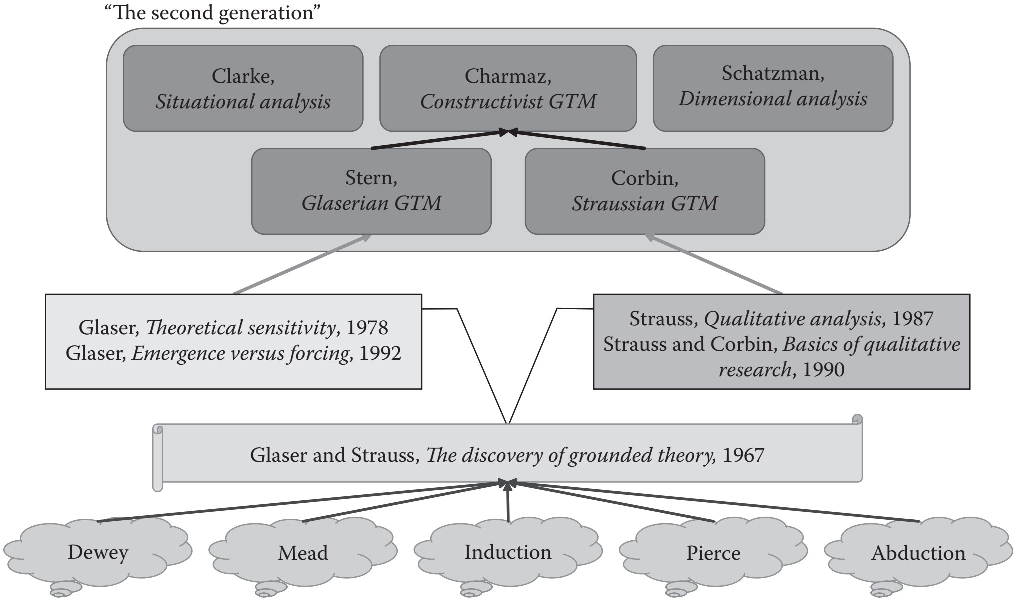

"The second generation"
Clarke, *Situational analysis*
Charmaz, *Constructivist GTM*
Schatzman, *Dimensional analysis*
Stern, *Glaserian GTM*
Corbin, *Straussian GTM*
Glaser, *Theoretical sensitivity*, 1978
Glaser, *Emergence versus forcing*, 1992
Strauss, *Qualitative analysis*, 1987
Strauss and Corbin, *Basics of qualitative research*, 1990
Glaser and Strauss, *The discovery of grounded theory*, 1967
Dewey
Mead
Induction
Pierce
Abduction

FIGURE 44.3 Diversity in grounded theory method.

published as Corbin and Strauss 2008)—a work that has sometimes been called "the cookbook" by Strauss's students and *their* students. Strauss's work in methods—especially the series of methods revisions with Corbin—has been extremely influential, and has allowed researchers with no direct ties to Strauss or his students to become competent practitioners of GTM.

Glaser considered these finely detailed methods to be a form of interference between the researcher and the data, and critiqued the Straussian procedures as "forcing" the data into a straitjacket imposed by arbitrary procedures (summarized in Glaser 1992). Glaser's perspective focused intensely on matters of "sensitivity" and "emergence" (Glaser 1978, 1998). A researcher who had attained theoretical sensitivity (Glaser 1978) would learn to attend to the data, and the theory would "emerge" from the data directly to the researcher, without being "forced" through unnecessary Straussian procedures (Glaser 1992).

These differences of emphasis led to increasingly divergent practices. Strauss developed detailed coding strategies and a vocabulary of coding methods and even of the kinds of notes that researchers keep about their data (see Section 44.5.2), as well as recommendations about how to fit a grounded theory analysis into the prior research literature. By contrast, Glaser's emphasis on working *only* from the data until the theory *emerges* from the data, led to a strong rejection of specific steps or procedures, and an insistence that the research literature should be avoided during the early stages of a project, as more of a source of distortion than of clarity. Table 44.1 summarizes these and other differences between the two approaches.

Students of Strauss, and students of Glaser, learned two increasingly divergent sets of concepts and methods, while leaders of each school continued to claim that it was the correct form of Grounded Theory. Understanding and resolving the differences in perspectives became, for some, a critical problem (e.g., Kelle 2005, 2007; van Niekerk and Roode 2009). In their analysis, van Niekerk and Roode (2009) suggested that the two versions of GTM could look very similar to a novice researcher. Focusing on the difficult process of *conceptualizing theory*, they contrasted the procedural approach of Corbin and Strauss (2008) with a greater flexibility and creativity in the approach of Glaser (1978, 1992, 1998). In their view, flexibility and creativity are an advantage, but only for the experienced researcher. Unfortunately, mentors who follow Glaser's approach are few and far between (see also Simmons [1994]; Stern [1994]). The Corbin and Strauss approach (2008), and the syntheses of Charmaz (2006a) and of Clarke (2005), offer stronger guidance for people who need to use grounded theory in the absence of a mentor. For this *Handbook* chapter, we have relied more on those accounts than on Glaser's works, because Glaser wrote more as arguments to experts, or as supplements to a formal mentoring relationship (1978, 1992, 1998).

Some of the students made additional methodological refinements, and brought in additional philosophical

TABLE 44.1
Summary of Differences between Glaserian and Straussian Grounded Theory Method

Issue	Glaserian GTM	Straussian GTM
Who should try this?	Only researchers who have conceptual ability and theoretical sensitivity.	Anyone who is willing to learn the appropriate procedures.
Research question	None or very general. Learn from the data.	Begin with a question, and refine it through data and theorizing.
Use of the literature	After ground theory analysis is complete.	Before, during, and after analysis, as appropriate.
Theoretical sensitivity	Comes from immersion in the data.	Use and learn from appropriate methods and tools.
Questioning	Focus is always on the data.	Use the data to pose questions, and answer the questions through constant comparison with other data and new data.
Coding	Substantive coding (also called open coding). Open coding ceases as soon as a core concept emerges. Codes are then refit and refined around the emerging core concept.	Simultaneous and constantly compared open coding, axial coding, and selective coding. Categories and dimensions are integrated with one another, leading to identification of core concept, leading to further coding compared with core concept.
Causative model	None.	In some versions of the "cookbook," a generalized causative model is recommended.
Theory attributes	Parsimony, scope, and modifiability.	Detailed and dense, with full description of process to refine theory.
Theoretical outcome	Abstract conceptualization.	Full description.
Memos and diagrams	Ideas about categories and their relationships. Diagrams used as needed.	Memos help to develop abstract thoughts (theory) about the data. Different types of memos are recommended for different types of coding issues (open, axial, selective, etc.). Diagrams help to clarify relationships among categories.

Note: Because Glaser and Strauss pursued different approaches to grounded theory, there has been some controversy. Even the comparisons of their two approaches are somewhat controversial. For this summary, we draw on work by Bryant (2002, 2003), Bryant and Charmaz 2007), Charmaz (2006a, 2009), Corbin and Strauss (2008), Gasson (2004), Glaser (1978, 1992, 1998, 2003, 2004), Heath and Cowley (2004), Kelle (2005, 2007), Morse et al. (2009), van Niekerk and Roode (2009), Strauss (1987, 1993), and Strauss and Corbin (1990, 1998).

and theoretical concepts. Stern (e.g., 2007, 2009) pursued Glaser's approach, while Corbin (2009; Corbin and Strauss 2008) pursued Strauss's approach. Others developed their own innovations, many of which are summarized in Bryant's and Charmaz's *Sage Handbook of Grounded Theory* (2007).

44.3.2 Steps toward a Unified Practice

A group of noted GTM students gathered together as a self-described "second generation" of ground theorists, and published a set of essays and method descriptions, along with transcripts of their discussions, as they attempted to bring the diverse strains of GTM back into a common discourse space: *Developing grounded theory: The second generation* (Morse et al. 2009). Among the "second generation," Charmaz's work on constructivist grounded theory (2006a,b, 2008, 2009) has become influential as both a source of procedural advice and a statement of the responsibility of the grounded theorist for the theory that he or she reports.

In Charmaz's constructivist synthesis of Glaser and of Strauss, the work of the grounded theorist goes beyond "forcing" and "emergence" to become a conscious *construction* of theory, always guided by data and disciplined by a set conceptual procedures that add rigor, credibility, and accountability of the theorist to the data at every step of the analysis (Charmaz 2006a, 2008; for a partially convergent perspectives, see Bryant [2002, 2003]; Clarke [2005, 2009]; Seale [1999]). The synthesis in our chapter is based primarily on the "cookbook" by Corbin and Strauss (2008) and the procedural descriptions of Charmaz (2006a), as well as pertinent observations by Glaser (1978, 1992, 1998), and illuminated by the work of other grounded theorists as appropriate.

44.4 WORKING WITH DATA IN GTM

There are three major aspects of GTM: working with data, working with theory, and structuring a GTM research project. In practice, a grounded theorist has to think about all three aspects in parallel. It is easier to write and read about the aspects separately, and we will therefore present each subtopic in its own section. We will integrate the three aspects as we go.

As outlined above, GTM is deeply concerned with data at every stage of the analysis. It seems fitting to begin with methods for working with the data. Corbin and Strauss (2008) and Charmaz (2006a,b) outline a similar set of practices for working with data. To make the concepts concrete we will illustrate them with examples from several different research programs (Allen 2003; Charmaz 2006), including some of our own (Muller, Millen, and Feinberg 2009; Thom-Santelli, Muller, and Millen 2008).

44.4.1 Describing the Data

The core work of grounded theory is to describe the data and the domain of the data through a series of descriptive *codes*. The descriptions are both highly detailed and specific to the domain, but also become more generalized and applicable to other domains as well (Figure 44.2). Star (2007) wrote, "A code sets up a relationship with your data, and with your respondents…. a matter of both attachment and separation…. Codes allow us to know about the field we study, and yet carry the abstraction of the new." Writing descriptions that are both accurately detailed and powerfully abstract is challenging. Kelle (2007) comments,

> Glaser and Strauss's initial idea that categories would emerge from the data if researchers with sufficient theoretical sensitivity would apply a technique of constant comparison was difficult to realize in practice. Consequently, this idea was modified and refined several times in the ongoing development of grounded theory leading to a variety of different, new, and complex concepts like *theoretical coding, coding families, axial coding, coding paradigm*, and many others that supplemented and sometimes displaced the concepts of constant comparison and theoretical sampling from the early days.

Influential accounts of GTM provide a four-step process to help to meet the challenge of how to get started in coding (Charmaz 2006a,b; Corbin and Strauss 2008; Dick 2005; Star 2007): open coding, axial coding, selective coding, and the designation of the core concept.

44.4.1.1 Open Coding

Coding of data begins with writing simple descriptive labels. This process is usually called "open coding." Initially, open coding is done by creating labels for the persons, objects, or concepts in each item of data (typically, an interview transcript, or a paragraph in an interview). Over time, certain codes (labels) begin to recur, and the researcher can begin to keep a list of recurring codes. A number of software applications are available to help with keeping a list of codes, and with applying codes directly from the list to the data.*

There has been much discussion on the granularity of coding. For example, in a study of configuration management, Allen (2003) attempted to follow the recommendations in the second edition of Strauss and Corbin (1998) for coding by "microanalysis which consists of analyzing data word-by-word…" (Figure 44.4). He found this level of analysis to be too time-consuming, and to lose the pattern of the data in too much irrelevant detail. Allen switched to a procedure he called "key-point coding," in which he first excerpted useful sentences or statements, and then applied codes against those large units of analysis (Figure 44.5).

Our own work has used open coding to identify topics of interest. In a study of employees who create named "collections" of files in a social file sharing system (Muller, Millen, and Feinberg 2009), we conducted interviews via instant messaging, and we coded the text that was recorded

* Because both authors work for a large company that provides software, we are ethically bound not to comment on other companies' software products. Therefore, with regret, we will not provide information about the available commercial tools that can be used to support grounded theory coding, or other aspects of GTM. Interested readers can use words from this chapter as search terms to begin to find these tools and to find blogs and articles that review these tools.

Informant Statement	**Open Code**
From my perspective	• *Personal view*
the main challenge is	• *Assertion*
in changes in technology	• *Changes in technology*
or the product improvement	• *Changes in product*
done by the … supplier.	• *Supplier*
You	• *Pronoun shift*
can never guarantee that	• *Assertion uncertainty*
if you are buying several	• *Procurement*
they will all be the same.	• *Product inconsistency*
	• *Necessary condition*

FIGURE 44.4 Microanalysis coding from a study of configuration management. (Excerpted from Allen, G. 2003. *Electron J Bus Res Methods* 2(1):1–10.)

Informant Statement	**Open Code**
Status accounting is used to report monthly to the Project Board.	• *CM process*
Main difficulty is in getting people to buy-in to CM.	• *People difficulty*
3rd parties have a preconceived set of established tools and are not willing	• *People difficulty*
to see the in-house point of view	• *Tool difficulty*
Developers saw CM as a control mechanism rather than a helpful tool.	• *Not helpful*
	• *Control*
	• *People difficulty*

FIGURE 44.5 Key-point coding form a study of configuration management. (Excerpted from Allen, G. 2003. *Electron J Bus Res Methods* 2(1):1–10.)

in the saved chat-session logs. The "Open Code" column of Figure 44.6 provides an excerpt of open coding from one of those logs. For our purposes, coding at the level of each interchange (pair of turns) in the chat session allowed sufficient granularity without making the coding task—or its outcome—overwhelming.*

Many other discussions could be cited here (e.g., Charmaz 2006a,b; Corbin and Strauss 2008; Locke 2001). The choice of the granularity depends in part on the sheer quantity of the data. Small sets of data can conveniently be analyzed in great detail, almost as if one were performing a close reading or an explication de texte. Larger sets of data will necessarily require a more macro-level focus.

Initially, the labels in open coding are not part of an organized body of concepts. Organizing the open codes into more complex conceptual structures occurs in the next several steps.

44.4.1.2 Axial Coding

In axial coding, the researcher begins to find relationships among the open codes (Charmaz 2006a,b; Corbin and Strauss 2008). Some of the relationships lead to clusters of codes. Often, it is possible to name each cluster. A named cluster is often called a "category," and much of early coding in GTM is a search for powerful categories. The right-most column of Figure 44.6 provides examples of axial coding, in which the open codes are organized into more abstract conceptual categories (axial codes).

In some cases, the codes that are organized into categories are simple, mutually exclusive alternatives, such as color names. In other cases, the codes can be arranged in a sequence or along a scale of some sort, such as the stressfulness of life events. In the latter case, the category is often termed a "dimension," and another goal of early coding in GTM is to discover these dimensions and to arrange the open codes along the dimensions—a process that is sometimes called "dimensionalizing the category." A dimension is a powerful concept for organizing the data, and is a step toward the "abstraction of the new" that Star wrote about (2007).

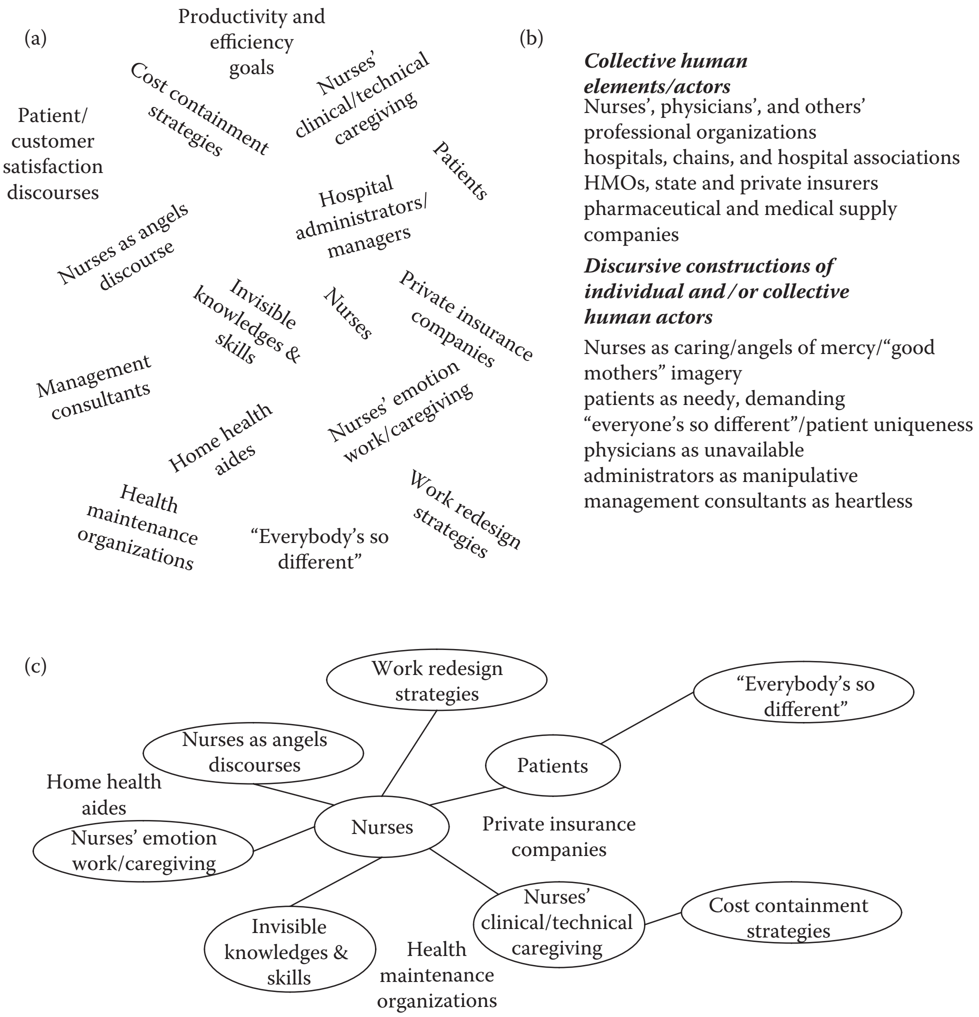
The relationship of axial codes to their constituent open codes can be described in several ways. Some grounded theorists prefer diagrams. In Allen's study of configuration management (Figures 44.4 and 44.5), he detailed the relationship of multiple open codes to a single axial code, as shown in Figure 44.7a and b (for reasons of space, we have presented only two of his axial codes, in summary form). Here, we can see a reuse of some of the open codes that were introduced in Figure 44.5, but now they have been renamed and organized in relation to one another. By contrast, we used *textual* representations, as shown in the right-most column of Figure 44.6 (Muller, Millen, and Feinberg 2009).

Several different forms of diagrams have been richly investigated by Clarke in her situational analysis revision of GTM (Clarke 2005, 2009), intended to be used after some basic coding has been done. Her *situational maps* (Figure 44.8) are

* Employees were aware that all chat-session logs could be recorded by any participant.

Chat question	Informant's Chat Answer	Open Code	Axial Code
Q. What was your goal (or goals) in using collections?	**A.** Put some structure around the content I collect/create around my topic for me and readers	*Structure around content* *For self* *For others*	*Purpose/structure content* *Self* *Audience*
Q. What kind of structure?	**A.** Taxonomy by topic, I guess	*Structure* *Taxonomy*	*Purpose/taxonomy*
Q. Did you make collections for yourself, and other collections for your readers? Or were all the collections for both "audiences"?	**A.** Both: what's good for me is good for my readers	*Collection for both self and others*	*Audience Self*
Q. Who are your readers?	**A.** *Sales teams, technical teams I do this basically for the sellers and supporting communities in the web 1.0 world I used teamrooms* *I needed an alternative*	*Readers* *Sales team* *Technical team* *Prior technology*	*Audience/Sales team* *Audience/Tech team* *Technology/team-room*

FIGURE 44.6 Open coding and axial coding from a study of collections in a social file sharing service. (Data from Muller, M. J., D. R. Millen, and J. Feinberg. 2009. Information curators in an enterprise file-sharing service. In *Proceedings of ECSCW 2009*, Springer, Vienna, Austria.)


(a)
Productivity and efficiency goals
Cost containment strategies
Nurses' clinical/technical caregiving
Patient/ customer satisfaction discourses
Patients
Hospital administrators/ managers
Nurses as angels discourse
Invisible knowledges & skills
Nurses
Private insurance companies
Management consultants
Nurses' emotion work/caregiving
Home health aides
Health maintenance organizations
"Everybody's so different"
Work redesign strategies
(b)
(c)
Work redesign strategies
"Everybody's so different"
Nurses as angels discourses
Patients
Home health aides
Nurses
Private insurance companies
Nurses' emotion work/caregiving
Nurses' clinical/technical caregiving
Cost containment strategies
Invisible knowledges & skills
Health maintenance organizations

Collective human elements/actors
Nurses', physicians', and others' professional organizations
hospitals, chains, and hospital associations
HMOs, state and private insurers
pharmaceutical and medical supply companies

Discursive constructions of individual and / or collective human actors
Nurses as caring/angels of mercy/"good mothers" imagery
patients as needy, demanding
"everyone's so different"/patient uniqueness
physicians as unavailable
administrators as manipulative
management consultants as heartless

FIGURE 44.7 Situational maps. (a) "Messy" situational map. (b) "Ordered" situational map. (c) Relationship map. Clarke also makes use of social worlds/arenas maps as a complex pattern of Venn diagram overlaps (not shown). (Redrawn from Clarke, A. E. 2005. *Situational Analysis: Grounded Theory after the Postmodern Turn*. Thousand Oaks, CA: Sage.)

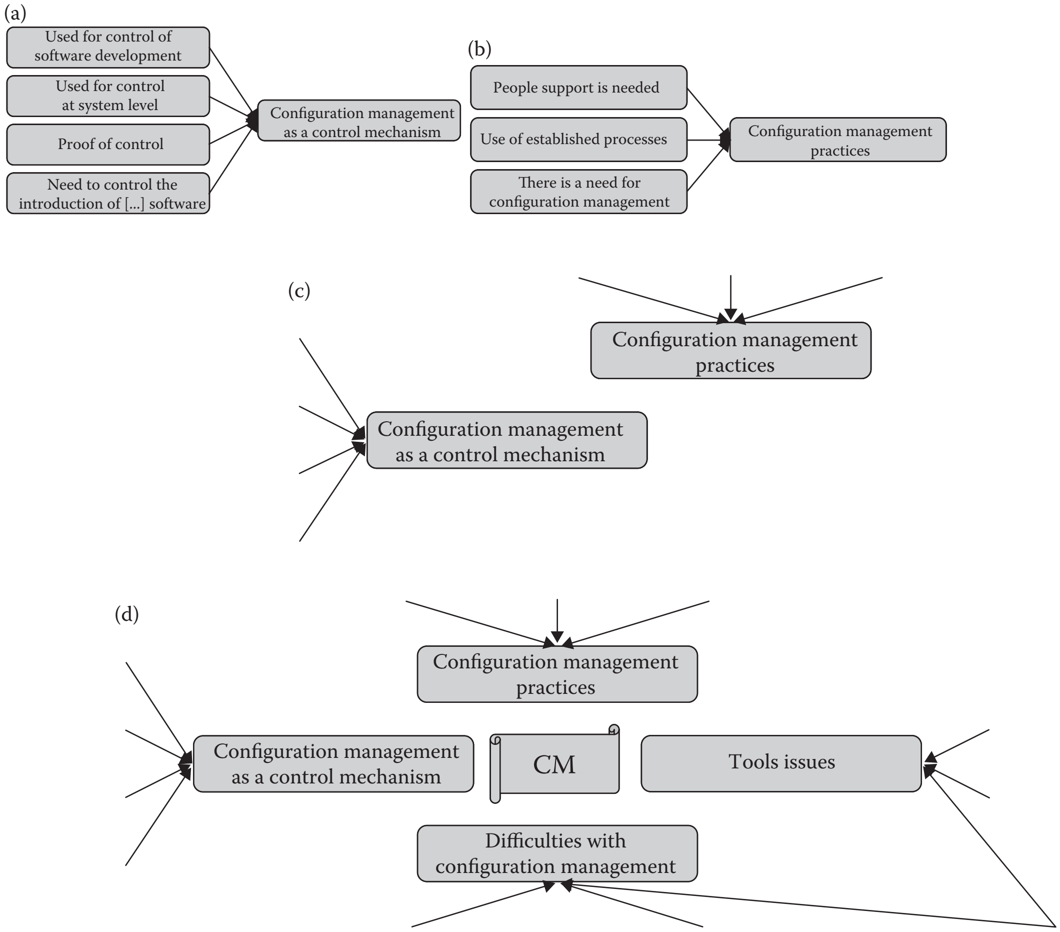

(a)
Used for control of software development
Used for control at system level
Proof of control
Need to control the introduction of [...] software
Configuration management as a control mechanism
(b)
People support is needed
Use of established processes
There is a need for configuration management
Configuration management practices
(c)
Configuration management practices
Configuration management as a control mechanism
(d)
Configuration management practices
Configuration management as a control mechanism
CM
Tools issues
Difficulties with configuration management

FIGURE 44.8 Diagramming in a grounded theory model study of configuration management. (a) and (b) Two axial codes. (c) The emergence of a topic (selective coding). (d) The core concept. (Redrawn from Allen, G. 2003. *Electron J Bus Res Methods* 2(1):1–10.)

similar to the diagrams of axial codes presented earlier (e.g., Figure 44.7), insofar as they can present a number of concepts in relation to one another, usually in a superset-subset visual organization. Clarke works back and forth between "messy" and "ordered" situational maps, as illustrated (by excerpt) in Figure 44.8. In the language of the preceding discussion, we interpret that the "messy" situational map contains primarily the open codes, while the "ordered" situational map organizes the open codes under axial codes.

Once a category or a dimension has been determined, the researcher may need to return to the data and recode the data in terms of the emergent concept that is summarized in the category or the dimension. For example, in the study excerpted in Figure 44.6, we had anticipated coding in terms of *Self* versus *Other*, but we had to return to each chat transcript in order to rework those open codes into the axial code of "Audience," and to catch all of the different types of audiences. While this return to the data takes time, it is an essential step in knowing the data well. In the language of Glaser, the constant return to the data is part of developing sensitivity, and allows the emergence of the insights that become theory.

Most projects are complex, and contain multiple categories and/or dimensions. The process described above—clustering open codes via axial coding, followed by recoding—may have to be followed multiple times, once for each axial code.

44.4.1.3 Selective Coding

Eventually, the researcher determines that some axial codes are more important than others. The researcher can then begin to focus on those sets of codes, and can begin to ignore the other, less important codes. By selecting certain codes for further development (Glaser, 1978, 1992; Corbin and Strauss 2008), the researcher is making a decision about what topics to pursue. When Charmaz (2006a,b) writes about *constructing* grounded theory, she is noting the importance of the researcher's choice in deciding what to focus on.

According to the standard sources for GTM, there are several ways to make this decision (Charmaz 2006a,b; Corbin and Strauss 2008). One way is to see which codes occur more often, or with greater generality across many different informants or different situations (Figure 44.8c presents a good example, in which the concept of "Nurses" is a strong organizing principle). A second way to do this is to see which codes occur in some contrasting pattern, for example, certain codes occur primarily with informants from one situation or attribute, while other codes occur primarily with different informants from a second situation or attribute (our *self-versus/and-audience* category from Figure 44.6 is a good example). A third way to do this is to note more complex patterns, or to understand that a particular category or dimension has explanatory power across many different situations or attributes. Each researcher will develop her or his own additional heuristics for deciding what is important.

Another way to determine importance is to consider broader, more formal theories that are *external* to the research project. This strategy was discouraged in the initial presentations of GTM (e.g., Glaser and Strauss 1967), and is anathema to the strict Glaserian school of GTM (e.g., Glaser 1992, 1998). However, HCI and CSCW make distinct demands upon researchers and research reports, including the requirement to explicate relevance to other work in the field. In our disciplines, it may be appropriate to consider formal, external theories at an earlier point in the research than in the "pure" grounded theory methodology from sociology and nursing.

Yet another possible heuristic is to consider the organizational setting in which the researcher is working. Grounded theory was developed in an academic setting, where researchers are often able to choose their GTM research topics based primarily on the data. By contrast, a GTM research project in an organization (Locke 2001) may be expected to produce answers to certain questions. During selective coding, it may be appropriate to choose topics based on those organizational priorities (e.g., what does the researcher's organization need to learn?).

In Allen's study, configuration management began to appear as a recurrent theme, as shown in Figure 44.7c. The two axial codes (Figure 44.7a and b) pointed toward the overall area of configuration management. However, other open codes and axial codes could have led in a different direction. Perhaps in part because of organizational priorities, Allen chose to narrow his focus to configuration management.

In our study of "curators" in file sharing, we began to see several key categories—*self-versus-audience*, and *self-sourced-versus-added-by-others*. Those two concepts emerged from the data. In addition, our previous GTM research in social tagging had led to similar categories of *audience* and also *impression-management* (see also Thom-Santelli, Muller, and Millen [2008]; Thom-Santelli, Cosley, and Gay [2006]). Based on the strong *audience*-related theme that we found in the initial file sharing codes, we used the external information from the social tagging study as an additional way to search for open and axial codes in the file sharing study. This step also required some amount of reviewing and recoding. The themes were indeed present in many of the interviews, and so we began to code selectively on themes of *self-versus/and-audience*, *source-of-file*, and *impression-management* around the role of the collection creator.

44.4.1.4 Core Concept and Closure

Finally, toward the end of the coding process, the researcher has to ask himself or herself, "What is this data a study of?" (Glaser 1978; also Kaufman 1986). What is the topic of this research? Once determined, the focal topic is often referred to as the "core concept" of the project and its report, and the process of determining the core concept is referred to as "closure." The answer to this question is found in the codes that were revealed through the previous step of selective coding—including the contextual constraints that were discussed in the Subsection 44.4.1.3 (e.g., need for relevance to other theories; fulfillment of organizational responsibilities).

How does the researcher determine the core concept? Opinions vary. In Charmaz's approach to GTM (2006a,b), the construct of the core concept is exactly that—a *construct* that is created as a human choice by the researcher, who then bears the responsibility for that act of construction. In Clarke's situational analysis, the accountability is also on the researcher, including a responsibility to ask (and to design the study to ask) *whose voice is not being heard (and why)? Whose silence is significant (and why)?*

By contrast, Glaser would claim that there is a kind of *inevitability within the data* to determine the core concept—that the core concept "emerges" after the researcher has spent sufficient time working with the data (e.g., Glaser 1992). Corbin and Strauss have tended to present a broader analytic framework that conveys its own *inevitability through the analytic process* leading to the core concept; at least one version of this analytic framework was entitled "The Paradigm" in earlier editions of their procedural work (e.g., Strauss and Corbin 1990).

In Allen's study, the choice made in selective coding led directly to the core concept of configuration management (Allen 2003).

In our study of file sharing, several interviews provided a key additional insight into how the file collectors viewed their own role, as shown in Figure 44.9a. All three of these informants appear to be engaged in an ongoing process of selecting files for their collections, with an explicit concern about how other people would be able to use those files (*collecting for others/audience)*. In addition, these and other informants also thought about how to name and describe their collections for the easiest discovery and use by others (Figure 44.9b).

These insights came from five informants in five different countries, and in four distinct job roles. Following the GTM principles of *theoretical sampling* (see Section 44.5.1), we tentatively concluded that this curator/editor code was found broadly across our sample. We also thought that the axial codes we had discovered earlier—*self-versus/and-audience* and *source-of-file*—were convergent with the concept of curating/editing. In addition, we realized that many people in a role similar to curators or editors would want to be known for their work, and that therefore the axial code of *impression-management* was also relevant. We conducted a last review of open codes and axial codes, and we reread selected chat sessions, and finally decided to construct our report around the concept of "curators."

44.4.1.5 Issues in Choosing the Core Concept

Is a core concept really required? Heuristically, most grounded theorists have recommended choosing a single topic around which to base an analysis, and choosing that topic as early as possible in the project (e.g., Corbin and Strauss 2008; Glaser 1978, 1992; Glaser and Strauss 1967; Locke 2001; Morse et al. 2009; Stern 2007; Strauss 1987, 1993). It is usually helpful for

A. Collecting for others

"regular collections with manually selected/curated resources.... trying to help people (and myself!) make sense of the files that are available.... putting together a collection and deciding what goes into it… and if they are different from the ones I've seen before then I add them to my collection…"(I15, enterprise 2.0 evangelist, Canada)

"a kind of editor, you share you own and other useful info via collections" (I18, sales, Finland)

"put some structure around the content I collect/create around my topic… what is good for me is good for my readers" (I19, product manager, France, already quoted in Figure 5).

B. Making the collection findable and usable

"very short descriptions… the intent of the collection – so I can keep the collection name really short!" (I9, project management, UK)

"sometimes I used the [descriptive field] to link to other related content [cross] reference" (I19, product manager, France)

and "i asked everyone to use the naming convention, and I enforced it" (I22, sales, USA)

FIGURE 44.9 Excerpts from the study of curators in a file-sharing service. (a) Code: *collecting for others*. (b) Code: *making the collection findable and usable*.

readers and reviewers if they know (to repeat Glaser's question [1978]) "What is this data a study of?"

But is this focus a *requirement* of grounded theory methodology? Or is it just a heuristic, or a rhetorical move, to increase the likelihood of publication? In their revision of GTM called "dimensional analysis," Bowers and Schatzman (2009) recommend delaying the choice of the core concept, and possibly keeping multiple core concepts throughout the later phase of the project. This is a minority view among grounded theorists, but it is available as a framework for researchers with complex domains.

In our discussion of the "uses of GTM in HCI and CSCW" (Section 44.1.2.1), we noted that some HCI researchers use grounded theory primarily for data analysis, and not to organize an entire project. In this case, the grounded theory part of the analysis is encapsulated within the broader project, and the "outcomes" of the grounded theory analysis may be partially structured by the requirements of that encapsulating project. For example, in a study of online communities (unpublished), we conducted a series of quantitative analyses of the social dynamics of more than 8000 distinct communities, and we used GTM to understand the attributes of a smaller number of communities that we had selected for detailed study. The results of the grounded theory analysis were then interpreted in the context of the broader quantitative study. In this case, the overarching themes of the research came from the quantitative study as well as the organization's needs, and we entertained multiple "core-concept-equivalents" as we developed and reported our grounded theory analyses. Other research projects have made similar uses of GTM to understand user comments or interviews within the context of a broader, multiple-method research project.

44.5 WORKING WITH THEORY IN GTM

Section 44.4 discussed how to code data, once the data have been collected. We also wrote, in the Section 44.1.2.1.2, that GTM advises to collect a small amount of data, and immediately begin coding and theorizing from that small sample. How is this done?

44.5.1 Theoretical Sampling through Constant Comparison

Outside of GTM, the conventional experimental approach favors either (1) large samples with high internal homogeneity or (2) stratified samples in which a particular variable is systematically varied (e.g., income or gender). The goal is to have sufficient data be able to characterize *all* the members of the large, homogeneous group, or *all* the members of each of the "strata" that make up the stratified sample (e.g., a homogeneous subset of high-income people, a homogeneous subset of medium-income people, etc.). By contrast, the usual initial strategy in GTM is to *sample for diversity* (e.g., Corbin and Strauss 2008). The assumption is that most interesting phenomena will become visible in the different ways that they occur across the diversity of people and organizations. Our observation of those differences will lead us to the factors that cause the differences, or at least the factors that are associated with the differences. Each set of observations is kept deliberately small, to allow for sampling across a greater diversity of people and situations.

44.5.1.1 Initial Sample

Remembering that the goal is to sample for diversity, the immediate question is, *How to start?* To choose the first sample of data, the researcher may try to find the most "representative" case or situation. Alternatively, the researcher might begin at an extreme of the range of possible person attributes or situations. For example, in our study of file sharing, we began by interviewing people who had used the "collection" features more than most other users. This initial sampling strategy assured us that we would receive a lot of information about collecting, but it also postponed interviews with people who were less frequent users. We made this choice, assuming that we would sample from the mid-frequency users later on.

We wanted to avoid any other source of bias in our sample, so we structured our list of high-frequency users in frequency order, and then we selected the informants such that they spanned as broad a range of attributes as possible. We interviewed at most one person from each country, and we tried to choose informants from as many different organizations as possible. This approach had the added advantage of avoiding the researcher's tendency to study other researchers. Using our criterion of different organizations, we would interview fewer members of the research organization, despite the ease of finding and recruiting them.

44.5.1.2 Subsequent Samples

In GTM, the first sample leads to the first set of codes, and then the first theory. The first theory allows us to make a quick, informal hypothesis—often of the form that, *if the phenomenon occurs under these circumstances, then it should (or should not) occur under a predictable second set of circumstances.* This hypothesis tells us how to choose our next sample: The second sample should abductively test the initial hypothesis. If the phenomenon occurs according to the hypothesized pattern, then that outcome strengthens the initial theory. If the phenomenon does not occur as predicted, then that tells us a limit to the initial theory, or tells us to recast the initial theory, or to replace it with a very different hypothesis.

44.5.1.3 Theoretical Sampling

The selection of the second (and subsequent) samples is made on the basis of the developing theory. For this reason, grounded theorists speak of *theoretical sampling.* In each case, the goal is to provide the strongest test *against* the developing theory (abduction), to reveal its weaknesses early, and to broaden the range of situations and attributes over which the theory makes good predictions or descriptions. As Charmaz writes, "Consistent with the logic of grounded theory, theoretical sampling is emergent. Your developing ideas shape what you do and the questions you pose while theoretical sampling" (Charmaz 2006a,b).

44.5.1.4 Constant Comparison

Theoretical sampling is a process of creation and destruction. What survives is a stronger theory, and a more informative—more powerful—way of describing the data. To continue Star's statement (2007) that we quoted in "Describing the Data,"

> Codes allow us to know about the field we study, and yet carry the abstraction of the new… When this process is repeated, and constantly compared across spaces and across data… this is known as theoretical sampling… Theoretical sampling stretches the codes, forcing other sorts of knowledge of the object… taking a code and moving it through the data… fractur[ing] both code and data.

Another way to describe this process is to say that data are always at the heart of ground theory, and that theory grows only by being tested against the data. Theory emerges *from the data* that we already have, and the emerging theory provides guidance about where or how to collect *the next set of data*. In summary, theory is repeatedly compared with old and new data. And new data are repeatedly compared with old data. As Star wrote, this is the principle of *constant comparison*, and lies at the heart of the inference process in GTM.

44.5.2 Memo-Writing and Writing from Memos

In keeping with the incremental nature of theoretical sampling, most grounded theorists write their emergent theory incrementally as well. In GTM, this process is called memo-writing, and is the recorded manifestation of the inference process, answering questions such as "what do I think is going on here?" and "what are the next data that I need to collect to test my theory?" and "what have I learned from the new data, and how do I code what I have learned?" and "what do I need to do next?" In the language of Section 44.5.1.3, "Memos are excellent source of directions for theoretical sampling—they point out gaps in existing analyses and possible new related directions for the emerging theory" (Glaser 2004). Consequently, memo-writing is begun as soon as the first data are collected, and continues throughout the project (Corbin and Strauss 2008).

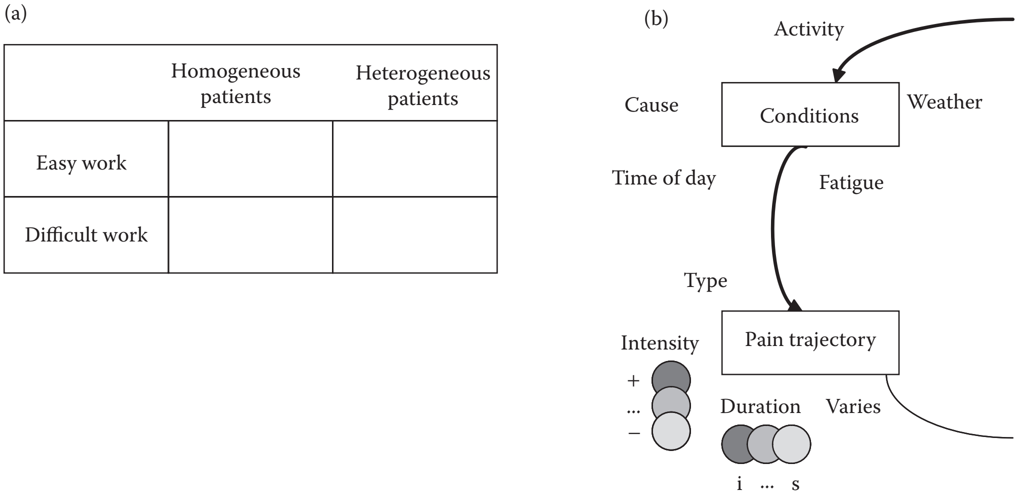
Memo-writing is considered to be a necessary component in conducting a GTM project:

> Theory articulation is facilitated through an extensive and systematic process of memoing that parallels the data analysis process in GT. Memos are theoretical notes about the data and the conceptual connections between categories. The writing of theoretical memos is the core stage in the process of generating theory.

(Glaser 2004). Corbin and Strauss (2008) concur: "[Memos] force the analyst to work with ideas instead of just raw data. Also, they enable analysts to use creativity and imagination, often stimulating new insights into data." Charmaz (2006a,b) agrees: "Memo-writing constitutes a crucial method in grounded theory because it prompts you to analyze your data and codes early in the research process…. [N]ote where you are on firm ground, and where you are making conjectures. Then go back to the field to check your conjectures."

Memo-writing goes on simultaneously with data collection and coding. In the memos, the researcher writes her or his justification for the codes, and makes a note of potential alternatives, hypotheses not yet tested, and implications (Glaser 2004). Stern writes (2007), "If data are the building blocks of the developing theory, [then] memos are the mortar." Charmaz adds (2006a,b), "You write memos throughout your research. Memos provide ways to compare data, to explore ideas about the codes, and to direct further data gathering. As you work with your data and codes, you become progressively more analytic in how you treat them and thus you raise certain codes to conceptual categories."

According to Glaser (2004), part of what makes memo-writing valuable is that it prevents jumping to conclusions, and permits thought: "Memos slow the analyst's pace, forcing him/her to reason through and verify categories and their integration and fit, relevance and work for the theory. In this way, he/she does not prematurely conclude the final



(a)

	Homogeneous patients	Heterogeneous patients
Easy work		
Difficult work		

FIGURE 44.10 Diagrams in memos. (Redrawn from Corbin, J., and A. L. Strauss. 2008. *Basics of Qualitative Research*. 3rd ed. Thousand Oaks, CA: Sage.)

theoretical framework and core variables." Corbin and Strauss agree about the importance of memo-writing: "Doing memos and diagrams should never be viewed as chores, or as tasks to be agonized over… [M]emos… grow in complexity, density, clarity, and accuracy as the research progresses… They move the analysis forward and as such are just as important to the research process as data gathering itself."

44.5.2.1 Forms and Genres of Memos

Memos can be long or short. Corbin and Strauss (2008) provide examples of memos related to many distinct perspectives and topics in a GTM investigation; not surprisingly, the length and structure of the memos vary from one-to-two paragraphs to a page or more. Charmaz (2006a,b) provides examples of memos that range from single paragraphs to well-structured essays, with explicit structuring of those essays using headers and sub-headers. By contrast, Dick (2005) writes that he always carries a supply of file cards, so that he can write memos as they occur to him; we infer that most of his memos are briefer than what can fit on a file card. Zaltman and Coulter (1995) used vignettes, and sometimes images. In our own research, our memos have ranged from writing multiple, numbered brief lines scribbled on a single file card, one memo per line, to a three-page, formally structured table that required several hours to produce with a word processor.

Specific memo-writing practices vary widely. Some researchers keep their memos in an organized set of files online. A few researchers record them sequentially in a notebook. Still other researchers write simple or elaborate diagrams (e.g., expanded versions of Figures 44.7, 44.8, and 44.10), or use software applications to generate diagrams from hierarchically structured codes (from open codes to axial to selective to core concept). For collaborative analyses, we often share memos online as documents in a social software application that supports collections of documents with controlled access to each collection. In addition to many textual examples, Corbin and Strauss (2008) described simple tables to explicate the relationship among multiple factors (Figure 44.10a), and causal diagrams to show hypothesized conceptual relationships (Figure 44.10b).*

Some researchers write specific genres of memos, while others do not differentiate. Using a memo classification scheme from Strauss and Corbin (1990), Jaccard and Jacoby (2010) described types of memos as follows:

> Researchers write down field notes to themselves about ideas and insights…which they then consult when analyzing their data, or when they formally posit their theory. One type of memo is called a *code memo*, which is a note relevant to the creation of coding or categories. A *theoretical memo*, by contrast, focuses on theoretical propositions linking categories or variables. *Operational memos* contain directions about the evolving research design and data collection strategies. Memo-writing occurs in the field during data collection and also during data analysis.

In their third edition revision, Corbin and Strauss (2008) proposed six general types of memos, but with less formality in the differentiations: open data explorations; identification and development of categories and dimensions; comparisons and questions; explorations among categories, dimensions, concepts; and development of a story line. By contrast, for Gasson (2004), all memos are "theoretical memos." As with much of the details of GTM, how the researcher adapts GTM to his or her own work is an emergent and highly personal process.

44.5.2.2 Connections through Memos

Stern's statement—that memos can serve as the "mortar" in constructing a grounded theory account (2007)—can be applied in several ways. First, memos can be used to summarize and integrate the findings from categories (Figure 44.11a) or

* However, the notebook format would interfere with another important attribute of memos according to Corbin and Strauss (2008), such that they would be capable of being physically sorted and rearranged. See Section 44.5.2.3. The use of online tools may provide most of the advantages of notebooks (dated, sequential entries) with sort-ability and may facilitate sharing and coordination as well.

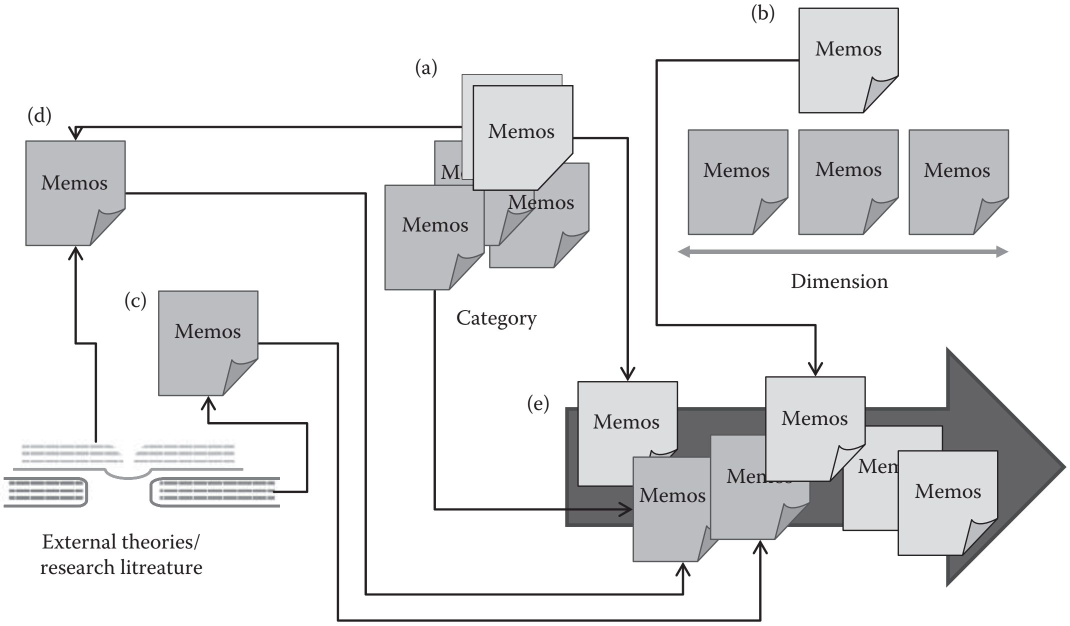


FIGURE 44.11 Memos in grounded theory model. (a) Memos based on category derived from data. (b) Memos based on dimension derived from data. (c) Memo from research literature. (d) Memo from data and research literature. (e) One prescription for writing the report based on the GTM investigation.

from dimensions (Figure 44.11b), that is, from concepts that originate in the data. In addition, as proposed by Glaser (2004) and as described in Pryor's thesis (2005), memos can serve as "mortar" between data and *external* information (Figure 44.11c), and between data and the research literature (Figure 44.11d). Memos provide an opportunity to try out different ways of exploring the meaning of data, or of connecting one item of data with another, or of looking for relationships among data, external knowledge, and the research literature. Some memos turn out to be valuable in report writing, while others can be "carried forward" into subsequent projects.

44.5.2.3 Writing from Memos

As we wrote, above, memos have much intrinsic value to the researcher during the process of doing the research, such as coding clarification; theoretical sampling, exploration of hypotheses; recording of paths not taken, and general support for the human thought and construction that is the core GTM activity. Memos are also valuable in the process of writing interim reports (which may stand as memos by themselves) and more formal results.

Of course, most researchers keep notes. What is special about memos in GTM? Grounded theory memos span predictable ranges, from instance to generalization, from concrete to abstract, from factual to theoretical (see Section 44.5.2.1). Over time, each concept of interest will have multiple memos related to it. When the researcher collects the relevant memos, he or she can assemble them into structure that "tells the story" or "makes the argument" about the relevant concept. Memos, thus, are both a concurrent tool for thinking, and also serve as an investment of sorts for later work, including report writing—as Glaser (2004) says, "a memo fund that is highly sort-able."

If, as Star said, "Theoretical sampling stretches the codes… fractur[ing] both code and data" (Star, 2007), then

> Sorting is essential—it puts the fractured data back together [because] the outline for writing is simply an emergent product of the sorting of memos…. Sorting… has a conceptual, zeroing-in capacity. The analyst soon sees where each concept fits and works, its relevance and how it will carry forward in the cumulative development of the theory.

(Glaser 2004). Glaser (2003) recommends finding a large surface and literally sorting the memos into the writing order for the report (Figure 44.11e). AlKaissi (n.d.) says, perhaps optimistically, "The sort structure is the report structure. It is often just a matter of preparing a first draft by typing up the cards in sequence and integrating them into a coherent argument." However, Stern (2007) cautions,

> The idea is to make labels act as rubrics for all known categories and their properties. Life isn't that tidy, and neither is memo sorting: it turns out that new labels are needed as categories collapse upon one another, and memos turn out to be misfiled and belong to another category.

In this necessarily challenging process, "Sorting helps the analyst integrate the theory; in the physical display of their thought processes, the appearance of the theory begins to take shape" (Stern 2007).

Memo-writing practices have implications for sorting. Dick (2005) and AlKaissi (n.d.) advocate writing memos on cards (e.g., file cards), which are easy to sort on a large table. By contrast, the longer memos described and shown by Charmaz (2006a) and Corbin and Strauss (2008) do not lend themselves to simple physical sorting: Most of these memos contain more than one idea. The diagrams-based memos (Figures 44.7, 44.8, and 44.10) that have been advocated by Clarke (2005) and Corbin and Strauss (2008) are unlikely to appear in their entirety and their multiplicity in a final report. The three-page table that we referred to from our own work presents similar problems.

Like so much of GTM, there is agreement in the overall concept of *writing from memos*, but each researcher will have to find his or her own set of personal best practices. In theoretical terms, memos can be used as a kind of externalization or crystallization of the researchers' thoughts (as in activity theory, e.g., Bazerman 1997), or can serve as a kind of extension of the researcher' cognition (as in distributed cognition, e.g., Ackerman and Halverson 2000; Hollan, Hutchins, and Kirsh 2000). Each researcher thinks differently from other researchers, and our work practices are different, and we think in a social context which may have its own requirements for information sharing at various stages of the grounded theory project (e.g., a researcher working alone as contrasted with a student being advised by a mentor). Some researchers will develop memo-writing strategies that are structured in terms of the eventual sorting process (e.g., AlKaissi, n.d.; Dick 2005; Glaser 2004). Other researchers will choose to make selected memos serve as collaborative documents, or highly specific and tightly focused interim reports on a particular concept (Charmaz 2006a; Corbin and Strauss 2008). Still other researchers will write memos containing first drafts of paragraphs, or even sections for their reports, or combinations of these practices.

44.6 STRUCTURING A GROUNDED THEORY MODEL RESEARCH PROJECT

Earlier, we wrote that there were three simultaneous components in a grounded theory project: working with data, working with theory, and structuring the project. We now turn to the third of those topics.

Grounded theory projects are both easier and more difficult to manage than conventional projects. The ease of management comes from being able to pursue new directions, and to use the researcher's developing knowledges and skills to follow-up on insights and possibilities. As we have said, grounded theory is as much about *thinking* as it is about the data, and it provides the thoughtful researcher with abundant opportunities to think long and hard and creatively and responsibly.

The difficulty of management comes from the potentially unbounded structure of the work. In a conventional, hypothetico-deductive project, the researcher designs a study with a known number of factors, a known number of informants, a fixed (and carefully planned) set of procedures that result in a set of data whose structure is known in advance, and a nearly automatic way of using the data to answer the research question. We know how to conduct these studies, and we know when each step is completed. By contrast, a grounded theory project has no "stopping rule" to tell us when data collection is complete, and when theorizing is complete. How do grounded theory researchers manage this problem?

One of the important concepts in grounded theory is "saturation" or having "saturated categories" (Charmaz 2006a; Clarke 2005; Corbin and Strauss 2008; Glaser 1978, 1992; Glaser and Strauss 1967; Strauss & Corwin 1990, 1998). The development of category or a dimension, from open codes through to selective coding, is a matter of going from concrete data to abstract concepts. Eventually, each new item of data produces less and less change in those abstract concepts—they become stable (a mathematician might say that they "reach asymptote"), and the categories on which they are based become fully connected with other categories and the emergent (Glaser and Strauss 1967) or constructed Charmaz 2006a) core concept. Because the categories are fully informed by data, and no longer develop, they are said to be "saturated." In a study of family violence, Stern (2007) describes the researcher's experience of saturation very persuasively:

> I realized that I had reached the point of saturation when the [informant] was telling me how when he was a small child he stood witness as his mother shot his father dead, *and I was bored*. I made all the right noises…but I knew that my data collection for that study had come to an end.

Another way to think about saturation is to refer back to the concept of abduction. As Reichertz (2007) reminds us, abduction is concerned with *surprise*, and with the structuring of a research project such that we are constructively and creatively surprised by new data and new concepts. When the surprise stops happening—when we are bored—then abduction is no longer taking place, and it is time to stop collecting data (and constructing theory), and time to start writing the report.

The process that we have just outlined is a "perfect" sequence for an academic research project with an unbounded timeframe and infinite resources. In practice, the researcher might face various constraints regarding time, access to informants, and budget—especially in an organizational setting. The stopping rule in the "perfect" academic setting might be "saturation," but in an organizational context, the stopping rule might very well turn out to be "Friday," or "when the budget is exhausted." As Charmaz (2008) reminds us, research takes place in a cultural and an organizational context. The researcher who adopts grounded theory in an organization may need to adapt some of the practices of grounded theory to meet the requirements of that organization.

44.7 APPLYING GROUNDED THEORY TO HUMAN–COMPUTER INTERACTION AND COMPUTER-SUPPORTED COOPERATIVE WORK

Grounded theory began in sociology, and has gained strength in many research disciplines, especially nursing. How can grounded theory be integrated within research and practice in HCI and CSCW?

44.7.1 Use of the Research Literature

A recurring problem in grounded theory is *What weight to give the research literature?* Early work (e.g., Glaser and Strauss 1967) took a firm stand for naïve investigations. Students were urged to postpone reading the research literature in order to remain open to the data, and only the data. This approach has turned out to be problematic in many ways. In the context of HCI and CSCW, research is required to make a novel contribution. We cannot know if our project or problem is novel, unless we have read extensively in the research literature.

There are deeper problems with the position of naïveté, *within* the process of conducting a grounded theory investigation. Kelle (2007) criticized this position as follows:

> The request 'literally to ignore the literature of theory and fact on the area under study, in order to assure that the emergence of categories will not be contaminated' (Glaser and Strauss 1967…) can lead inexperienced users of grounded theory procedures to adopt an unrealistic idea about their work. Novices who wish to firmly observe the principle of 'emergence' often experience the search for categories as extremely tedious and a subject of sometimes numerous and endless team sessions, leading to a proliferation of categories which makes the whole process insurmountable …

How, then, can the research literature enter a grounded theory investigation without distorting the process? Dey (1993) claimed that "there is a difference between an open mind and an empty head. To analyze data, we need to use accumulated knowledge, not dispense with it." Discussing Dey's work, Vasconcelos (2007) adds "In other words, analysis is emergent, but not 'atheoretical'" (see also Goulding [1998]). In HCI and CSCW, this point is particularly important, both in making the argument from the data, and in demonstrating the value of the research as a novel contribution that answers questions relevant to the research literature in our fields.

In Figure 44.2, we offered a summary diagram of how a grounded theory investigation proceeds—from a broad view of the data to a narrower focus, and from a shallow set of categories and dimensions to a richer, deeper, more interconnected set of concepts. In grounded theory research, the theory that emerges (or is constructed) *from the data* is called "substantive theory," whereas concepts from the research literature are called "formal theory." The question thus becomes, *When and how should substantive theory be informed by formal theory?*

44.7.1.1 Sensitizing Concepts

Grounded theorists have discussed an approach to this problem by reference to Blumer's earlier proposal of the "sensitizing concept" (Blumer 1954). A sensitizing concept is to be used as "a starting point in a qualitative study" to "draw attention to important features… and provide guidelines… in specific settings…" (Bowen 2006). Charmaz describes sensitizing concepts as "background ideas that inform the overall research problem" (see also Charmaz [2003]; Glaser [1978]; Vasconcelos [2007]). Earlier in this chapter, we claimed that grounded theory takes advantage of the human tendencies to think and to theorize. In keeping with that claim, we note Bowen's quotation of Gilligan's argument that "Research usually begins with such concepts, whether researchers… are aware of them or not."*

The principled use of sensitizing concepts helps us to be aware of how we think and theorize in this way, and to understand where we are led by the data (in the purer sense of grounded theory) and where we are influenced by the work that has gone before our own research. Based on Blaikie (2000), Bowen (2006) suggests that sensitizing concepts be used *not* as a source of hypotheses to be tested, but rather "to lay the foundation for the analysis" or "to develop thematic categories" from data that have already collected, perhaps then helping to choose the next set of data to be (iteratively) collected.

44.7.1.2 Coding Families

A second strategy for making carefully limited use of the research literature was proposed by Glaser (1978), as a set of "coding families." Glaser offered his coding families in the form of established sets of axial codes, based in broad theoretical backgrounds in sociology, and ready to be applied to data as needed. Coding families include concepts such as causality, limit or range of values, part-whole relations, and collective beliefs. These selected types of relationships, collected by a seasoned researcher, may be one way to inform the work with an attenuated set of results from the research literature: "The conception of coding families makes clear that certain types of theoretical knowledge are clearly helpful in deriving grounded categories from the data" (Kelle 2007).

44.7.1.3 Summary

Our impression is that grounded theorists continue to struggle (as we do) with the "right" balance between maintaining an "open mind," while avoiding the problems associated with an "empty head" (Dey 1993). Researchers in HCI and CSCW will need to inform their choice of research area and problem to be solved with a strong knowledge of the research literature; too much naïveté is likely to lead to repeated work and missed opportunities. What appears to be crucial, is to remain aware of when we are working from the research literature, and when we working from the data, and to make thoughtful choices about how to mix the two perspectives. Discussions of this question, written in the context of HCI and CSCW, would be very helpful, and we hope our colleagues will address these issues.

44.7.2 Presentation of Results and Implications

The fields of HCI and CSCW have their own norms of how arguments are made, and how reports are structured. If a grounded theory project is to be published in these literatures, then of course it must be "rhetorically recognizable,"

* Gilligan's source paper, referred to as a web page in Bower's paper, is no longer available online.

that is, it must present its method and its analysis in a rigorous manner, and it must make a case for the quality of the data reporting and analysis, and for the conclusions that are drawn from the data. We recommend that the researcher indicate clearly whose GTMs he or she has adopted, and (if appropriate) how he or she has adapted the method in response to local circumstances or organizational requirements. Most HCI and CSCW audiences will not be familiar with the special language of grounded theory, so we recommend that the researcher make careful decisions between necessary precision of GTM terms, versus burdening the reader with what may appear to be jargon. GTMs are intended to help the researcher to achieve clarity and precision of thought. The researcher has a responsibility to provide that precision and especially that clarity *to the reader*, in terms that the reader can understand and use in his or her own research.

44.8 CONCLUSION

GTMs are less than 50 years old, and are undergoing change, reformulation, innovation, and experimentation (e.g., Morse et al. 2009). The adoption of GTM in HCI and CSCW has begun even more recently, and researchers in our fields are continuing to learn about the methods, and about how they can be adapted to advantage under the various constraints of HCI and CSCW work in academia, industry, government, and nonprofits. In this chapter, we have presented a summary of our understanding of GTM in 2010, written with the intention that researchers can learn enough from these pages to begin their own practices (the learn-by-doing approach is advised by many of the sources on GT methods). We anticipate rapid growth in understanding of GTM in our fields within the next 5 years. We hope that this chapter helps to spur that growth, and that the new understandings will replace this chapter with a better set of advices and findings.

During this period of learning, growth, adoption, and adaptation, each researcher will be developing her or his own set of practices, and will be struggling with the conflicting advices of the many grounded theorists, as well as their diverse philosophical positions. As Floyd (1987) noted in her account of *process-oriented software* engineering, practices in our field are under development, just as software and hardware are under development. In this chapter, we have attempted to provide a broad survey of practices and the rationales behind them. We have noted the disagreements, while focusing on the commonalities, and we have provided pointers to discussions and divergent approaches that we hope will be useful as each HCI or CSCW researcher navigates this rich, promising, and challenging domain of analysis, thought, action, and passion.

Grounded theory has tended to be used in HCI and CSCW in two distinct ways. Perhaps the greater number of papers have used grounded theory as an analytic method on data that have already been collected. All of the coding methods and iterative theory-construction practices come into play here, except of course for the fullest extent of theoretical sampling (because, in most of these studies, the data set has already been collected, and there is limited or no opportunity to determine a need for more data from a previously unsampled source). We note that many HCI and CSCW web researchers currently face more data than any human can analyze, and that they can, to some extent, exercise theoretical sampling by choosing *which data to focus on* among the huge datasets that come to us in Internet or intranet sampling.

A numerically smaller number of HCI and CSCW projects has used a grounded theory approach that is closer to the "ideal" sociological methods from the 1967 *Discovery of Grounded Theory*, that is, the ability to collect data in small samplings, using constant comparison techniques in order to choose where, how, and with whom the next data iteration should be conducted.

There has been an extended discussion of issues of quality and rigor in GT (Corbin and Strauss 2008; Charmaz 2006a, 2009; Gasson 2004; Muller 2010), including several distinct positions about boundaries, that is, what kind of work qualifies as GTM (Adolph, Hall, and Kruchten 2008; Becker 1993; Cutliffe 2005; Matavire and Brown 2008; Skodal-Wilson and Ambler-Hutchinson 1996; van Niekerk and Roode 2009; and of course the ongoing disagreements by Glaser and by Strauss, cited earlier). In our survey of critiques of GT quality and rigor, we discovered some authors who claimed that the "full" or "ideal" version of GTM was the only valid way to conduct research that could be called "grounded theory." However, we think it is too early for anyone to say, for HCI and CSCW, what "is" or "is not" grounded theory. We make reference again to Floyd's (1987) insights about the development of *methodology* that accompanies the development of technology and of theory. Researchers in HCI and CSCW will continue to develop our fields' understandings of these methods, and as a community of researchers we will make new HCI- and CSCW-based innovations in grounded theory for our domains. Where appropriate, our community will bring our innovations to the sociologists and nursing researchers who have taught us so much about grounded theory, and inform their practices with insights from our own.

What we need now is continued growth and development of GTM in HCI and CSCW, but we principally need discussion. By analogy, it is time to begin to apply the grounded theory concept of "constant comparison" to our own fields, comparing our methods and our findings with other people in HCI and CSCW as a kind of "data about doing GTM research." We will rapidly form iterative theories about what works and what does not, and then we should abductively test those theories in further GTM-based studies. We hope to see not only more papers on grounded theory in HCI and CSCW, but also more workshops and panels, where we can perform a collective "constant comparison" of adaptations of GTM to our fields. To heal the schism and advance the field, Morse et al. (2009) wrote *Developing grounded theory: The second generation*. Perhaps we will soon be able to write and read a collection of papers titled, *Developing grounded theory in HCI and CSCW: The third generation*.

REFERENCES*

Ackerman, M. S., and C. A. Halverson. 2000. Reexamining organizational memory. *Commun ACM* 43(1):58–64.

Adolph, S., W. Hall, and P. Kruchten. 2008. A methodological leg to stand on: Lessons learned using grounded theory to study software development. In *Proceedings of Conference on the Center for Advanced Studies on Collaborative Research 2008*, Toronto, ON, Canada. New York: ACM.

AlKaissi, A. A. E. (n.d.). Grounded theory. An-Najah National University http://www.najah.edu/file/Essays/english/Aidah%20Kaisi/19.pdf.

Allen, G. 2003. A critique of using grounded theory as a research method. *Electron J Bus Res Methods* 2(1):1–10.

Asimakopoulos, S., R. Fildes, and A. Dix. 2009. Forecasting software visualizations: An explorative study. In *Proceedings of British Conference on HCI*, 269–77. Cambridge, UK. New York: ACM

Awbrey, J., and S. Awbrey. 1995. Interpretation as Action: The Risk of Inquiry. *Inquiry Crit Thinking Across Disciplines* 15:40–52.

Barr, P., J. Noble, R. Biddle, and R. Khaled. 2006. From pushing buttons to play and progress: Value and interaction in fable. In *Proceedings of Australian User Interface Conference 2006*, 61–8. Hobart, Australia. New York: ACM

Bazerman, C. 1997. Discursively structured activities. *Mind Culture Act* 4(4):296–308.

Becker, P. H. 1993. Common pitfalls in published grounded theory. *Qual Health Res* 3(2):254–6.

Bertram, D., A. Voida, S. Greenberg, and R. Walker. 2010. Communication, collaboration, and bugs: The social nature of issue tracking in small, collocated teams. In *Proceedings of CSCW 2010*. Savannah, GA. New York: ACM.

Blaikie, N. W. H. 2000. *Designing Social Research: The Logic of Anticipation*. Cambridge, UK: Polity.

Blumer, H. 1954. What is wrong with social theory? *Am Soc Rev* 18:3–10.

Blythe, M., and P. Cairns. 2009. Critical methods and user generated content: The iPhone on YouTube. In *Proceedings of CHI 2009*, 1467–76. Boston, MA. New York: ACM.

Bowen, G. A. 2006 Grounded theory and sensitizing concepts. *Int J Qual Methods* 5(3):12–23.

Bowers, B., and L. Schatzman. 2009. Dimensional analysis. In *Developing Grounded Theory: The Second Generation*, ed. J. M. Morse, P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. Walnut Creek, CA: Left Coast Press.

Bryant, A. 2002. Regrounding grounded theory. *J Inf Technol Theory Appl* 4:25–42.

Bryant, A. 2003. Doing grounded theory constructively. A reply to Barney Glaser. *Forum Qual Sozialforschung/Forum Qual Soc Res* 3(3).

Bryant, A., and K. Charmaz, eds. 2007. *The Sage Handbook of Grounded Theory*. Thousand Oaks, CA: Sage.

Charmaz, K. 2003. Grounded theory: Objectivist and constructivist methods. In *Strategies of qualitative inquiry 2nd Edn*, ed. N. Denzin and Y. Lincoln 249–91. Thousand Oaks, California: Sage Publications.

Charmaz, K. 2006a. *Constructing Grounded Theory: A Practical Guide through Qualitative Analysis*. Thousand Oaks, CA: Sage.

Charmaz, K. 2006b. Grounded theory. In *Encyclopedia of Sociology*, ed. G. Ritzer. Cambridge, MA: Blackwell.

Charmaz, K. 2008. Grounded theory in the 21st century: Applications for advancing social justice. In *Strategies of Qualitative Research,* ed. N. K. Denzin and Y. S. Lincoln, 3rd ed., 203–41. Thousand Oaks, CA: Sage.

Charmaz, K. 2009. Shifting the grounds: Constructivist grounded theory methods. In *Developing Grounded Theory: The Second Generation*, ed. J. M. Morse, P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. Walnut Creek, CA: Left Coast Press.

Chiovitti, R. F., and N. Piran. 2003. Rigour and grounded theory research. J *Adv Nurs* 44(4):427–35.

Clarke, A. E. 2005. *Situational Analysis: Grounded Theory After the Postmodern Turn*. Thousand Oaks, CA: Sage.

Clarke, A. E. 2009. From grounded theory to situational analysis: What's new? Why? How? In *Developing Grounded Theory: The Second Generation*, ed. J. M. Morse, P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. Walnut Creek, CA: Left Coast Press.

Corbin, J., and A. L. Strauss. 2008. Basics *of Qualitative Research*. 3rd ed. Thousand Oaks, CA: Sage.

Corbin, J. 2009. The Straussian perspective. In *Developing Grounded Theory: The Second Generation*, ed. J. M. Morse, P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. Walnut Creek, CA: Left Coast Press.

Cutliffe, J. R. 2005. Adapt or adopt: Developing and transgressing the methodological boundaries of grounded theory. *J Adv Nurs* 51(4):421–8.

Dagenais, B., H. Ossher, R. K. E. Bellamy, M. P. Robillard, and J. P. deVries. 2010. Moving into a new project landscape. In *Proceedings of Conference on Software Engineering 2010*, 275–84. Cape Town, South Africa. IEEE.

Dalsgaard, T., M. B. Skov, M. Stougaard, and B. Thomassen. 2006. Mediated intimacy in families: Understanding the relation between children and parents. In *Proceedings of Conference on Interface Design and Children 2006*, 145–52. Tampere, Finland. New York: ACM.

Dewey, J. 1986. Logic: The theory of inquiry. In *John Dewey: The Later Works, 1925–53.* (Vol. 12: 1938), ed. J. A. Boydston. Carbondale, IL: Southern Illinois University Press.

Dey, I. 1993. *Qualitative Data Analysis: A User-Friendly Guide for Social Scientists*. London: Routledge.

Dick, B. 2005. *Grounded Theory: A Thumbnail Sketch*. Resource papers in action research, http://www.scu.edu.au/schools/gcm/ar/arp/grounded.html.

Dodig-Crnkkovic, G. 2002. Scientific methods in computer science. In *Proceedings of Conference for Promotion of Research in IT*, 446–60. Skovde, Sweden. Scientific Research Publishing.

Eiter, T., and G. Gottlob. 1995. The complexity of logic-based abduction. *J ACM* 42(1):3–42.

Faisal, S., B. Craft, P. Cairns, and A. Blandford. 2008. Internalization, qualitative methods, and evaluation. In *Proceedings of BELIV 2008*. Florence, IT. New York: ACM.

Ferreira, J., J. Noble, and R. Biddle. 2007. Up-front interaction design in agile development. In *Proceedings of Conference on Agile Processes in Software Engineering and Extreme Programming 2007,* 9–16. Como, IT. Springer.

Fincher, S., and J. Tenenberg. 2007. Warren's question. In *Proceedings of Workshop on Computer Education Research*, 51–60. Atlanta, GA. New York: ACM.

Floyd, C. 1987. Outline of a paradigm change in software engineering. In *Computers and Democracy: A Scandinavian Challenge,* ed. G. Bjerknes, P. Ehn, and M. Kyng. Brookfield, VT: Gower.

* All URLs listed in this chapter were verified during July 2010.

Gasson, S. 2004. Rigor in grounded theory research: An interpretive perspective on generating theory from qualitative field studies. In *The Handbook of Information Systems Research*, ed. M. Whitman and A. B. Woszczynski. Hershey, PA: Idea Group.

Ge, X., Y. Dong, and K. Huang. 2006. Shared knowledge construction process in an open-source software development community: An investigation of the Gallery community. In *Proceedings of Conference on Learning Sciences 2006*, 189–95. Bloomington, IN. International Society of the Learning Sciences.

Glaser, B. G. 1978. *Theoretical Sensitivity*. Mill Valley, CA: Sociology Press.

Glaser, B. G. 1992. *Basics of Grounded Theory Analysis: Emergence vs Forcing*. Mill Valley, CA: Sociology Press.

Glaser, B. G. 1998. *Doing Grounded Theory: Issues and Discussions*. Mill Valley, CA: Sociology Press.

Glaser, B. G. 2003. *The Grounded Theory Perspective II: Description's Remodeling of Grounded Theory Methodology*. Mill Valley, CA: Sociology Press.

Glaser, B. G. 2004. Remodeling grounded theory. *Forum Qual Soc Res Soz* 5(2).

Glaser, B. G., and A. L. Strauss. 1965. *Awareness of Dying*. Chicago, IL: Aldine.

Glaser, B. G., and A. L. Strauss. 1967. *The Discovery of Grounded Theory*. Chicago, IL: Aldine.

Glaser, B. G., and A. L. Strauss. 1968. *A Time for Dying*. Chicago, IL: Aldine.

Goede, R., and C. de Villiers. 2003. The applicability of grounded theory as research methodology in studies on the use of methodologies in IS practices. In *Proceedings of the 2003 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on Enablement Through Technology* 2003. South Africa: South African Institute for Computer Scientists and Information Technologists.

Goulding, C. 1998. Grounded theory: The missing methodology on the interpretivist agenda. *Qual Market Res* 1(1):50–7.

Greenberg, S., and H. Thimbleby. 1992. The weak science of human-computer interaction. In *CHI '92 Research Symposium on Human Computer Interaction*. Monterey, California. New York: ACM.

Haig, B. D. 2005. Grounded theory as scientific method. In *Philosophy of Education Yearbook 2005*. Philosophy of Education Society. University of Illinois.

Heath, H., and Cowley, S. 2004. Developing a grounded theory approach: A comparison of Glaser and Strauss. *International Journal of Nursing Studies* 41:141–50.

Hollan, J., E. Hutchins, and D. Kirsh. 2000. Distributed cognition: Toward a new foundation for human-computer interaction. *Trans Comput Hum Interact* 7(2):174–96.

Hood, J. C. 2007. Orthodoxy vs. power: The defining traits of grounded theory. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant, and K. Charmaz, 191–213. Thousand Oaks, CA: Sage.

Jaccard, J., and J. Jacoby. 2010. *Theory Construction and Model-Building Skills: A Practical Guide for Social Scientists*. New York: Guilford.

Jantunen, S., and K. Smolander. 2006. Towards global market-driven software development processes: An industrial case study. In *Proceedings of Workshop on Global Software Development for the Practitioner*, 94–100. Shanghai, China. New York: ACM.

Jin, J., and L. A. Dabbish. 2009. Self-interruption on the computer: A typology of discretionary task interleaving. In *Proceedings of CHI 2009*, 1799–808. Boston, MA. New York: ACM.

Kane, S. K., C. Javant, J. O. Wobbrock, and R. E. Ladner. 2009. Freedom to roam: A study of mobile device adoption and accessibility for people with visual and motor disabilities. In *Proceedings of ASSETS 2009*, 115–22. Pittsburgh, PA. New York: ACM.

Kaptelinin, V., B. Nardi, S. Bødker, J. Carroll, J. Hollan, E. Hutchins, and T. Winograd. 2003. Post-cognitivist HCI: Second-wave theories. In *CHI 2003 Extended Abstracts*. Fort Lauderdale, FL: ACM.

Kaufman, S. R. 1986. *The Ageless Self: Sources of Meaning in Late Life*. Madison, WI: University of Wisconsin Press.

Kelle, U. 2005. "Emergence" vs. "forcing" of empirical data? A crucial problem of "grounded theory" reconsidered. *Forum Qual Soc Res* 6(2):191–213.

Kelle, U. 2007. The development of categories: Different approaches in grounded theory. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant, and K. Charmaz, 191–213. Thousand Oaks, CA: Sage.

Kierkegaard, B., and P. Borlund. 2008. Characteristics of information needs for television broadcasts of scholars and students in media studies. In *Proceedings of Symposium of Information Interaction in Context 2008*, 116–22. London, UK. New York: ACM.

Lee, G., W. DeLone, and J. A. Espinosa. 2006. Ambidextrous coping strategies in globally distributed software development projects. *Commun ACM* 49(10):35–40.

Lempert, L. B. 2007. Asking questions of the data: Memo writing in the grounded theory tradition. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant and K. Charmaz, 245–64. Thousand Oaks, CA: Sage.

Locke, K. 2001. *Grounded Theory in Management Research*. Thousand Oaks, CA: Sage.

Mackay, W. E., and A.-L. Fayard. 1997. HCI, natural science and design: A framework for triangulation across disciplines. In *Proceedings of DIS'97*, 223–34. Amsterdam, The Netherlands: ACM.

Mamykina, L., A. D. Miller, E. D. Mynatt, and D. Greenblatt. 2010. Constructing identities through storytelling in diabetes management. In *Proceedings of CHI 2010*, 1203–12. Atlanta, GA. New York: ACM.

Matavire, R., and I. Brown. 2008. Investigating the uses of "grounded theory" in information systems research. In *Proceedings of SAICSIT 2008*, 139–47. Wilderness. South Africa. New York: ACM.

Menzies, T. 1996. Applications of abduction: Knowledge-level modeling. *Int J Hum Comput Stud* 45(3):305–35.

Morgan, L., and P. Finnegan. 2010. Open innovation in secondary software firms: An exploration of managers' perceptions of open source software. *ACM SIGMIS Database* 41(1):79–95.

Morse, J. M., P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. 2009. *Developing Grounded Theory: The Second Generation*. Walnut Creek, CA: Left Coast Press.

Muller, M. J. 2010. Grounded theory method. Tutorial at HCIC 2010. Slides http://www.slideshare.net/traincroft.

Muller, M. J., D. R. Millen, and J. Feinberg. 2009. Information curators in an enterprise file-sharing service. In *Proceedings of ECSCW 2009*, Springer, Vienna, Austria. Springer.

Nørgaard, M., and K. Hornbæk. 2006. What do usability evaluators do in practice? An explorative study of think-aloud testing. In *Proceedings of DIS 2006*, 209–18. University Park, PA. New York: ACM.

Patokorpi, E. 2009. What could abductive reasoning contribute to human computer interaction? A technology domestication view. *PsychNology J* 7(1):113–31.

Peirce, C. S. 1865/1982. On the logic of science. In *Writings of Charles S. Peirce: A Chronological Edition,* ed. Peirce Edition Project, Vol. 1, 1857–66. Bloomington, IN: Indiana University Press.

Popper, K. 1968. *The Logic of Scientific Discovery*. 2nd ed. New York: Harper Torchbook.

Pryor, J. 2005. *A Grounded Theory of Nursing's Contribution to Inpatient Rehabilitation*. PhD thesis. School of Nursing, Deakin University.

Reichertz, J. 2007. Abduction: The logic of discovery of grounded theory. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant and K. Charmaz, 214–28. Thousand Oaks, CA: Sage.

Sanderson, P. M. 2003. Cognitive work analysis. In *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science*, ed. J. M. Carroll, 225–64. San Francisco, CA: Morgan Kaufman.

Sarker, S., F. Lau, and S. Sahey. 2001. Using an adapted grounded theory approach for inductive theory building about virtual team development. *Data Base Adv Inf Syst* 32(1):38–56.

Sarker, S. 2006. Knowledge transfer and collaboration in distributed U.S-Thai teams. *Journal of Computer Mediated Communications* 10(4).

Savago, S., and J. Blat. 2009. About the relevance of accessibility barriers in the everyday interactions of older people with the web. In *Proceedings of Cross-Disciplinary Conference on Web Accessibility 2009*, 104–13. Madrid, Spain. New York: ACM.

Schulte, C., and M. Knobelsdorf. 2007. Attitudes towards computer science-computing experiences as a starting point and barrier to computer science. In *Proceedings of Workshop on Computing Education Research 2007*, 27–35. Atlanta, GA. New York: ACM.

Seale, C. 1999. *The Quality of Qualitative Research*. London, UK: Sage.

Sease, R., and D. W. McDonald. 2009. Musical fingerprints: Collaboration around home media collections. In *Proceedings of GROUP 2009*, 331–40. Sanibel Island, FL. New York: ACM.

Simmons, O. E. 1994. Grounded therapy. In *More Grounded Theory Methodology: A Reader*, ed. B. Glaser, 4–37. Mill Valley, CA: Sociology Press.

Skodal-Wilson, H., and S. Ambler-Hutchinson. 1996. Methodological mistakes in grounded theory. *Nurs Res* 45(2):122–4.

Sousa, C. A. A., and P. H. J. Hendriks. 2006. The diving bell and the butterfly: The need for grounded theory in developing a knowledge based view of organizations. *Organ Res Methods* 9(3):315–38.

Star, S. L. 2007. Living grounded theory. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant and K. Charmaz. Thousand Oaks, CA: Sage.

Stern, P. N. 1994. Eroding grounded theory. In *Critical Issues in Qualitative Research Methods*, ed. J. Morse. Thousand Oaks, CA: Sage.

Stern, P. N. 2007. Properties for growing grounded theory. In *The Sage Handbook of Grounded Theory*, ed. A. Bryant and K. Charmaz. Thousand Oaks, CA: Sage.

Stern, P. N. 2009. Glaserian grounded theory. In *Developing Grounded Theory: The Second Generation*, ed. J. M. Morse, P. N. Stern, J. Corbin, B. Bowers, K. Charmaz, and A. E. Clarke. Walnut Creek, CA: Left Coast Press.

Strauss, A. L. 1987. *Qualitative Analysis for Social Scientists*. New York: Cambridge.

Strauss, A. L. 1993. *Continual Permutations of Action*. New York: Aldine.

Strauss, A. L., and J. Corbin. 1990. *Basics of Qualitative Research*. Thousand Oaks, CA: Sage.

Strauss, A. L., and J. Corbin. 1998. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. 2nd ed. Thousand Oaks, CA: Sage.

Strauss, A. L., and B. G. Glaser. 1970. *Anguish*. Mill Valley, CA: Sociology Press.

Suddaby, R. 2006. From the editors: What grounded theory is not. *Acad Manage J* 49(4):633–42.

Swallow, D., M. Blythe, and P. Wright. 2005. Grounding experience: Relating theory and method to evaluate the user experience of smartphones. In *Proceedings of EACE 2005*. Chania, Greece. University of Athens.

Taipale, O., and K. Smolander. 2006. Improving software testing by observing practice. In *Proceedings of Symposium on Empirical Software Engineering*, 262–71. Rio de Janeiro, Brazil. IEEE.

Thom-Santelli, J., D. Cosley, and G. Gay. 2006. What's mine is mine: Territoriality in collaborative authoring. In *Proceedings of CHI 2009*. Boston, MA. New York: ACM

Thom-Santelli, J., M. J. Muller, and D. R. Millen. 2008. Social tagging roles: Publishers, evangelists, leaders.' In *Proceedings of CHI 2008*. Florence, IT.

Umarji, M., and C. Seaman. 2008. Why do programmers avoid metrics? In *Proceedings of Symposium on Empirical Software Engineering and Measurement 2008*, 236–47. Kaiserslautern, DE. New York: ACM.

Umarji, M., and C. Seaman. 2009. Gauging acceptance of software metrics: Comparing perspectives of managers and developers. In *Proceedings of Empirical Software Engineering and Measurement 2009*, 236–47. Kaiserslautern, DE: Fraunhofer-Gesellschaft.

van Niekerk, J. C., and J. D. Roode. 2009. Glaserian and Straussian grounded theory: Similar or completely different? In *Proceedings of SAICSIT 2009*.

Vasconcelos, A. C. 2007. The use of grounded theory and of arenas/social worlds theory in discourse studies: A case study on the discursive adaptation of information systems. *Electron J Bus Res Methods* 5(2):125–36.

Whitworth, E., and R. Biddle. 2007. Motivation and cohesion in agile teams. In *Proceedings of Conference on Agile Processes in Software Engineering and Extreme Programming 2007,* 62–9. Como, IT. New York: ACM.

Williams, L., L. Layman, K. M. Slaten, S. B. Berenson, and C. Seaman. 2007. On the Impact of a Collaborative Pedagogy on African American Millennial Students in Software Engineering. In *Proceedings of Conference on Software Engineering 2007*, 677–84.

Yu, C. H. 1994. Is there a logic of exploratory data analysis? In *Annual Meeting of American Educational Research Association*. New Orleans, LA. American Educational Research Association. New York: ACM.

Zaltman, G., and R. Coulter. 1995. Seeing the voice of the customer: Metaphor-based advertising research. *J Adv Res* 83:35–51.

45 An Ethnographic Approach to Design

Jeanette Blomberg and Mark Burrell

CONTENTS

45.1 INTRODUCTION

In recent years, academic and professional researchers and designers working in the field of human–computer interaction (HCI) have looked to ethnography to provide a perspective on relations between humans and the artifacts and solutions they design and use.* Within the field of HCI there are different views among researchers and practitioners on just what constitutes an ethnographic inquiry. For some, ethnography is simply a fashionable term for any form of qualitative research. For others, it is less about method and more about the lens through which human activities are viewed. In this chapter, we will attempt to position the ethnographic approach within historical and contemporary contexts, outline its guiding principles, detail the primary methods and techniques used in ethnographically informed design practice, and provide case examples of ethnography in action.

This chapter provides an introduction to ethnography, primarily as it relates to studies in HCI. We will touch only briefly on some of the more controversial topics current within the field of ethnographic research that have enlivened mainstream academic discourse in recent years. We will point the reader to books and articles where these topics are discussed in more detail. Our primary aim in this chapter is to provide academics and professionals in the field of HCI with a working understanding of ethnography, an appreciation for its value in designing new technologies and practices, and a discerning eye when it comes to reviewing and evaluating ethnographically informed design studies.

45.2 THE RELEVANCE OF ETHNOGRAPHY FOR DESIGN

The turn to ethnography as a resource for design can be traced back to the early 1980s when computer technologies were moving out of the research labs and engineering environments and into mainstream office settings, call centers, manufacturing floors, and educational institutions. There was the realization that the designers and developers of these technologies could no longer rely exclusively on their own experiences as a guide for the user requirements of these new systems. Instead, designers and developers needed a way to gain an understanding of the everyday realities of people working within these diverse settings (Blomberg et al. 1991). In many organizations, market research groups were being asked to provide perspectives on the people and practices that made up these varied settings. However, the techniques most commonly used by market research groups at the time (e.g., attitude surveys, focus groups, telephone interviews, etc.) were not well suited for developing an actionable understanding of what people actually do day-to-day that could inform the design of new products, services and interactive solutions.

Anthropologists and other social scientists had long recognized that what people say and what they do can vary significantly, making reliance on surveys, focus groups, and telephone interviews insufficient for the task. Designers and developers needed a way to get a firsthand view of the on-the-ground realities—the “here and now”—of everyday life in these diverse settings. At this time in the early 1980s, social scientists working at the Xerox Palo Alto Research Center were beginning to explore ways of bringing insights from ethnographic research into a productive relationship with the design of new technologies (e.g., Blomberg 1987, 1988, 1995; Suchman 1983; Suchman, Blomberg, and Trigg 1999). Not long after, other research labs (e.g., Hewlett-Packard, Apple Computer, and NYNEX) followed suit (e.g., Nardi and Miller

* Ethnographic research is often just one of many approaches used to inform design. Usability studies, surveys, business case analysis, scenario planning, future workshops and social network analysis are a few of the approaches that are used in conjunction with ethnography.

1990; Sachs 1995). Today, many industrial research and development labs in the United States have anthropologists and other social scientists with ethnographic expertise on staff (e.g., IBM, Intel, Kodak, Microsoft, Motorola, General Motors, and Xerox, to name but a few).

Ethnographically informed design practices also began to take hold in design firms and consulting companies during the early 1990s (e.g., IDEO, Fitch, and the Doblin group). These early explorations led, in 1993, to the founding of e-Lab, a research and design company that distinguished itself from other design firms at the time by creating an equal partnership between research and design (Wasson 2000). Ethnographic methods were at the center of e-Lab's research approach, with a commitment to base design recommendations on insights from ethnographic research (Robinson 1994). Today there are a number of research and design firms who provide ethnographically informed design solutions (e.g. Adaptive Path, Cheskin, Continuum, GravityTank, HLB, IDEO, and Jump, to name a few).

Furthermore, in the mid-1980s the growth in networked applications and devices, made possible through the availability of local area networks and early Internet implementations, created awareness among designers and developers that they would need a strategy that focused beyond support for single, isolated users interacting with information technologies. They would need a way to understand the information and communication practices of people interacting with one another, both face-to-face and through mediating technologies. Information technologies were increasingly becoming communication and collaboration technologies that consequently demanded an examination of social interaction across time and space. In response, a group of computer scientists, human-factors engineers, and social scientists, somewhat dissatisfied with the dominant perspectives within HCI at the time founded the field of computer-supported cooperative work (CSCW)* (e.g., Grief 1988; Schmidt and Bannon 1992). A group of sociologists at Lancaster University and researchers at the Xerox Research Center in Cambridge, England played a prominent role in helping to shape the ethnographic research agenda within CSCW (e.g., Bentley et al. 1992; Hughes, Randall, and Shapiro 1993; Rodden and Anderson 1994; Hughes, Rodden, and Anderson 1995).

Finally, the explosion of the Internet in the late 1990s accelerated the move of information technologies out of the workplace and into homes, recreational environments and other non-work-related settings. This redoubled interest in the ethnographic perspective as a valuable tool in the design of new technologies and technology mediated services. This presented a new set of challenges for designers as they were asked to design and build applications that leveraged powerful, digital technologies for use by people of all ages, engaged in myriad non-work-related activities in diverse contexts. Although the clamor for all that is the Internet has somewhat subsided, the legacy of that period is that researchers and designers who learned their craft during the Internet boom years have gone on to positions in academia and industry, in both boutique design firms and major companies, and in a variety of industries including advertising, marketing, product development, and IT services. In late 2005, many in the ethnographic design community assembled at an industry sponsored conference, EPIC (Ethnographic Praxis in Industry Conference). The conference brought together a diverse group of researchers working in areas such as product design, workplace studies, and business ethnography to define the scope of a collective agenda and to strengthen professional ties and research connections (Anderson and Lovejoy 2005). This conference (see Cefkin [2009]; Proceedings of Ethnographic Praxis in Industry Conferences 2005–2011) continues today and is a powerful testament to the continuing value of focusing on people's everyday realities and experiences—the here and now—when designing innovative technologies, experiences, and services.

45.3 THE ROOTS OF ETHNOGRAPHY

Ethnography has its historical roots in anthropology, but today is an approach found in most all of the traditional and applied social sciences, and in interdisciplinary fields such as HCI and Human Factors Engineering. In anthropology, ethnography developed as a way to explore the everyday realities of people living in small-scale, non-Western societies and to make understandings of those realities available to others. The approach relies on the ability of all humans to decipher what is going on through participation in social life. The techniques of ethnography bear a close resemblance to the routine ways people make sense of the world in everyday life (e.g., by observing what others do, participating in activities, and talking with others). The research techniques and strategies of ethnography developed and evolved over the years to provide ways for the ethnographer to "be present" for the mundane, the exceptional, and the extraordinary events in people's lives.

More recently within the field of anthropology both the focus on non-Western peoples and the implicit assumptions made about non-Western societies (e.g., that they are bounded, closed, and somewhat static) have undergone a transformation. Today, the ethnographic approach is not limited to investigations of small-scale societies, but instead is applied to the study of people and social groups in specific settings within large industrialized societies, such as workplaces, senior centers, and schools, and specific activities such as leisure travel, financial investing, teaching, and energy consumption, to name but a few. Consequently, new techniques and perspectives have been developed and incorporated into anthropological and ethnographic inquiry. However, a few basic principles have continued to inform and guide ethnographic practice. These principles include

* The dominant perspectives at the time emphasized technological possibilities over the uses and users of technology, the interface requirements of stand-alone applications over networked devices, and human psychology and cognition over social interaction. However, by the late 1990s ethnographically informed design attained a prominent place in HCI research, and today there is considerable overlap between the fields of CSCW and HCI.

studying phenomena in their *natural settings*, taking a *holistic* view, providing a *descriptive understanding,* and taking a *member's perspective*.

45.4 PRINCIPLES OF ETHNOGRAPHY

45.4.1 Natural Settings

Ethnography is anchored in the underlying assumption that to gain an understanding of the world you must encounter it firsthand. As such, ethnographic studies always include gathering information in the settings in which the activities of interest normally occur. This does not mean that ethnographic studies never involve techniques that remove people from those everyday settings or that introduce into those settings artifacts or activities that would not be present otherwise. The insistence on studying activities in their everyday settings is motivated by the recognition that people have a limited ability to describe what they do and how they do it without access to the social and material aspects of their environments. Furthermore, people's ability to fully articulate what they do is constrained by the tacit understandings that guide actions (Polanyi 1966). Finally, some aspects of people's experiences are best studied by observing and recording the ongoing flow of activities as they occur (e.g., people's patterned movements through settings such as retail stores or airports, moment-by-moment shifts in scheduling, etc.).

45.4.2 Holistic

Related to the emphasis on natural settings is the view that activities must be understood within the larger context in which they occur. Historically within anthropology the notion of holism focused attention on the fact that societies were more than the sum of their parts (however these parts were specified). The particular aspects of a society (e.g., the court system) could only be understood in relation to the other aspects of the society (e.g., kinship system, belief system). Today, because ethnography is less often applied to the study of entire societies, the notion of holism has a somewhat different emphasis. Holism holds that studying an activity in isolation, without reference to the other activities with which it is connected in time and space, provides a partial and potentially misleading understanding of that activity. So, for example, investigating online search strategies without understanding how these strategies fit into the larger set of activities in which search is but one component (e.g., in the context of online trading, shopping, or report writing) narrows the possible insights from the study.

45.4.3 Descriptive

Ethnographic accounts have always provided a descriptive understanding of people's everyday activities. Ethnographers are concerned first and foremost with understanding events and activities as they occur, without evaluating the efficacy of people's everyday practices. This is not to say that ethnographic accounts cannot or should not be used to suggest how things could be different or to point out inequities or inadequacies in current ways of doing things. Indeed, as applied in the domain of HCI, ethnography is aimed at identifying opportunities for change. However, there is a strong conviction that to suggest changes or to evaluate a situation, one first needs to understand it as it is. The work practice and technology group at the Xerox PARC (Palo Alto Research Center) developed a slogan expressing this conviction that innovation requires an understanding of the present (Figure 45.1).

Innovation
=
Imagination of what could be
based in a knowledge of what is

FIGURE 45.1 Innovation.

As such, ethnographic accounts strive first and foremost to provide descriptive and not prescriptive understandings of people's everyday lives. In recent years, there have been many challenges to the idea that a purely descriptive understanding is possible. Critics point out that every account is shaped by the perspectives of the researcher, the goals of the project, and the dynamics of the relationship between the investigator and those studied, to name but a few factors that shape ethnographic accounts. While we do not argue with this position, we contend that the value of ethnography for design is not diminished by recognition that our accounts are always located and partial.

45.4.4 Members' Point of View

As already alluded to, ethnographers are interested in gaining an insider's view of a situation. They want to see the world from the perspective of the people studied and describe behaviors in terms relevant and meaningful to the study participants. As such, ethnographers are interested in the ways people categorize their world and in the specific language people use to talk about things. This perspective is sometimes at odds with the requirements of quantitative survey research in which the relevant categories must be known in advance of the study and in which the categories and the language used cannot vary across participant groups. In quantitative approaches, the terms and categories used are likely to be those of the research community and not those of the study participants, which can undermine the validity of the results (see Section 45.7 for further discussion of this topic).

45.5 THE POSTMODERN INFLECTION

The scientific paradigm within which ethnography evolved has come under serious questioning over the last quarter century as social studies of science have shown how scientific knowledge production is shaped by the larger social context in which scientific inquiries take place (Latour 1987; Latour and Woolgar 1986; Pickering 1980). As part of this critical discourse, ethnographic accounts have been challenged for their veracity. Likewise the authority of the ethnographic voice has been questioned (Clifford 1988; Clifford and Marcus 1986; Marcus and Fischer 1986). These challenges have come from a number of fronts, most significantly from study participants who increasingly are able to read ethnographic accounts (Said 1978) and from feminists who saw in many ethnographic accounts a Western, male bias (Harding 1986; Smith 1987; Wolf 1992; Yanagisako and Delaney 1995). These challenges have made researchers from all fields of inquiry more aware of how their research is shaped by the particular time and place in which it occurs. It is our view that knowledge of the world is always mediated by presuppositions, be they cultural, theoretical, or practical, and as such no ethnographic account is value-free. But we also contend that this does not diminish the value and efficacy of an ethnographic approach as a resource for designing new technologies, experiences, and services. Maintaining the illusion of a theoretically neutral and value-free absolute "truth" is not necessary to establish the value of ethnographic research in design. By striving to describe and understand how people operate in and construe their everyday "realities," ethnography can provide useful frameworks and roadmaps to guide the design of "people-centered" solutions.

45.6 ETHICAL ISSUES

As will be discussed in more detail later, ethnographic research requires developing the trust and participation of the people studied. Without this trust participants will be reluctant to allow researchers into their homes, boardrooms, and classrooms and they will not openly share their everyday experiences and concerns. Anthropologists have long realized that such a privileged, trusted position requires reciprocity—if you allow me access to your world, I will protect your interests. This bargain has not always been easy for ethnographers to keep. Over the years there have been examples of ethnographic research, where, wittingly or not, the situation of the people studied has been compromised.*

In the context in which ethnographic research is being used to inform the design of new products and services—ones that will change people's lives—it is critical that the ethnographer reflect on the impact this research could have on study participants. Of course, it is not possible to control all the ways findings from ethnographic research will be used, nor how innovations informed by these studies will be integrated into people's lives. But the ethnographer can work to protect study participants from immediate harm (e.g., that was caused by divulging a worker's identity to management) and can inform study participants of possible longer term negative impacts (e.g., job losses brought about by introduction of new technologies). As ethnographic research has moved into new contexts (e.g., HCI, organizational development), it has been necessary to think creatively about how our ethical guidelines map to these new conditions. However, we cannot lose sight of the importance of protecting the interests of those who have agreed to participate in our studies be they workers in organizations, traders on Wall Street, or mothers of special needs children.

45.7 ETHNOGRAPHIC METHODS

The ethnographic method is not simply a toolbox of techniques, but a way of looking at a problem, a "theoretically informed practice" (Comaroff and Comaroff 1992, quoted in Agar 1996:7). The methods and techniques outlined later in this chapter have been developed over the years to enable the development of a descriptive and holistic view of activities as they occur in their everyday setting from the point of view of study participants. We are not attempting to be exhaustive in our presentation, nor do we want to suggest that there is a fixed set of canonical ethnographic methods and techniques. We encourage researchers to continue developing new techniques as the circumstances require (e.g., studying "virtual" communities, globally distributed workgroups, technologically mediated interactions). What remains constant in the ethnographic approach is a commitment to describe the everyday experiences of people as they occur.

45.7.1 Research Planning

One of the keys to a successful research project is the creation of a plan of action to guide the research and support changes and adjustments that inevitably must be made as the project proceeds. Research planning can be divided into three general stages: (1) formulating research objectives, (2) devising a strategy for selecting study participants, and (3) selecting appropriate research techniques and approaches.

Research objectives follow from the specific questions to be addressed by the research. It can be useful to develop an explicit statement that clearly articulates the objectives of a given study. This statement acts as a beacon to help keep the research on track through the many twists and turns of a project. For example, if the research aims to inform the development of a software application that will help doctors manage patients' records, the research statement could be something as simple as "understand how doctors manage patient records through all the phases of treatment and in the varied settings in which they practice medicine." Over the course of a project, the research objectives' statement (along with the research design and plan) may change as a project team coalesces and learns about the experiences of the people in the particular domain of interest.

45.7.2 Study Participants

Once the research objectives have been identified, a strategy for selecting study participants (sometimes referred to as a "sampling strategy") is devised that answers two primary

* To mitigate such negative impacts, the American Anthropological Association has developed a code of ethics that provides guidance for people engaged in ethnographic research. This code outlines the appropriate disclosures and protections that should be given to study participants. (Fluehr-Lobban [1991] provides a discussion of ethical issues in anthropological research.)

questions: (1) what types of participants best suit the research objectives and (2) how many participants should be included in the study to achieve the research objectives? The strategy for selecting study participants is influenced by the research focus (e.g., shopping behavior vs. workgroup collaboration) and may include selecting at different levels of abstraction (e.g., which organizations, which workgroups, and which individual employees). In addition, as Cohen (2005) has cautioned, attention should be given in making these choices for those who might intentionally or inadvertently be excluded from the study and as such rendered invisible to the research lens.

Several types of sampling strategies are employed by social science researchers, which fall under two main categories: (1) probability and (2) nonprobability (Bernard 1995).[*] Our focus in this chapter is on nonprobability sampling, as that is most commonly employed in ethnographic research.[†] The nature of ethnographic work, as well as recruiting constraints often demand selecting participants based on criteria other than a strict probability.

Four types of sampling fall under the rubric of nonprobability: (1) quota, (2) purposive, (3) convenience, and (4) snowball (Bernard 1995). When sampling by quota, the researcher specifies which groups are of interest (e.g., women, teenagers, truck drivers, people who use software X, organizations with fewer than 100 employees, etc.) and how many will be needed in each group. The number of groups chosen will depend on the research objectives and the amount of time available, but the basic idea is to cover the range of possible variation one would expect across the target population. Practically speaking, when identifying the variables or factors that should be considered in sampling to enable visibility into possible variations in experiences and practices, the ethnographer will often presumptively identify "differences that may make a difference" in the experiential domain of inquiry. For example, if the focus is on how people manage their personal finances, the researcher might deliberately strive to specifically sample people with varied financial situations in addition to life/career stages or family situations. To ensure the desired variability is covered—particularly when the researcher is dependent on others to provide access to or recruit the participants for a study—it is useful to create a "screener,"[‡] a questionnaire-like instrument designed to identify characteristics that are appropriate for a given project. Quota sampling is only possible when the desired participants[§] are easy to identify in advance and recruit. If it is not possible or desirable to specify how many participants will be in each sampled group, a purposive sampling strategy may be called for. This sampling strategy is based on the same principles as quota sampling, but the number of participants for each group is not specified.

Convenience and snowball sampling rely on a "sample as you go" strategy. This is required in situations in which you do not know in advance who will be available to participate or which individuals or groups should participate. Convenience sampling entails selecting people who are available, meet the requirements of the research, and are willing to participate. One might use this strategy, for example, to observe and interview people as they shop in a grocery store.

Snowball sampling relies on participants referring others whom they think would be good candidates for the research, or on researchers identifying individuals or groups to be included in the study as the research proceeds. Because this method utilizes existing social networks it is especially valuable when desired participants are initially inaccessible or reluctant to participate (e.g., CEOs, drug users, club members) or when the relevant population cannot be known in advance.[¶]

45.7.3 Gaining Access

One of the challenges for ethnographic research is gaining access to field sites and study participants. Access to institutional settings often requires getting permission from management to observe and interview employees, or from school officials and parents to spend time in classrooms. In some cases, written permission that specifies certain terms and conditions (e.g., how confidential information will be protected) is required before researchers are allowed onsite. In other cases, recruiting agencies may be used to identify participants and financial incentives may be offered for participating in the study. The time (and skill) required to establish these initial relationships and agreements should not be underestimated.[**]

45.7.4 Observation

As discussed earlier, ethnographers are interested in understanding human behavior in the contexts in which it naturally occurs, making observation one of the hallmark methods of the approach. In academic settings, it has been common for anthropologists to spend a full year at a given field site. While

[*] The intent behind probability sampling, or statistical sampling, is to generalize from the research sample to a larger population with a specified degree of accuracy, measured in terms of probability. All types of probability sampling require a randomly selected and relatively large sample size.

[†] Using nonprobability samples does not mean we cannot make general statements. If participants are chosen carefully, one can obtain reliable data with as few as four or five participants (Nielsen and Landauer 1993). Additionally, a recent case study demonstrates that smaller, nonrandomly selected samples can produce the same results as large-scale survey research for as little as 1/100 of the cost (Green 2001). A nonprobability strategy also does not preclude conducting a statistical analysis or measuring differences between individuals or groups using nonparametric statistics, such as Fisher's Exact Test or nonparametric correlation measures. Their limitation is that they cannot be used to make claims about larger populations within a specified degree of probability.

[‡] Screeners are an essential tool if using an external recruiting agency to locate study participants.

[§] For sampling purposes, participants need not be individuals, but could be families, households, workgroups, or other naturally occurring entities.

[¶] Johnson (1990) provides a more detailed discussion of sampling in ethnography.

[**] Anthropologists have been accused in the past of only studying the disempowered and disenfranchised because these individuals were less likely to feel powerful enough to refuse participation in ethnographic studies. Although important in all contexts, when studying people with more power and ability to say no (Nader 1974), it is often necessary to demonstrate how their participation will be of benefit to them, their community or workplace, or the wider society.

this continues to be the case for more traditional ethnographic studies, shifts in research focus (e.g., away from studies of entire societies), and in study locations (e.g., away from isolated, hard to reach settings) have resulted in more varied research designs which may involve shorter, intermittent fieldwork periods in one or more locations. Moreover, in some applied settings (e.g., enterprise work environments) the time available for field observation may be constrained, sometimes allowing for no more than a few days in any one setting.

45.7.4.1 Why Observe?

One of the fundamental axioms in the social sciences, and anthropology in particular, is that what people say they do and what they actually do are frequently quite different. Studies have shown verbal reports to be inconsistent with observed behavior in a number of areas including (among many other examples) shopping behavior (Rathje and Murphy 1991), child rearing (Whiting and Whiting 1970), recycling (Corral-Verduga 1997), and health habits (Rich et al. 2000).

The discrepancies between verbal reports and behavior can be due to a variety of factors. People may be concerned with their image and so report, consciously or not, behavior that is more socially acceptable. Along these same lines, participants may respond to questions in a particular way in an attempt to please the researcher. Another source of disparity between behavior and verbal reports is that people are often not aware of their actual behavior because it is so habitual. Such tacit knowledge is often not easily accessible through interview techniques alone (D'Andrade 1995).

The limitation of human memory is another reason why interview data can differ from observations. When asking participants about past events, or recurring patterns of behavior, our memory may be selective and skew responses in any number of directions, sometimes in predictable patterns (Bernard 1995).

The complexity of social life is another reason individual accounts of an event may miss certain relevant details. The environments in which humans interact are extremely dynamic and complex—composed of social relationships, artifacts, and physical spaces—and making it difficult for individuals to fully envision, let alone articulate after the fact, what is going on.

45.7.4.2 The Researcher's Observational Role

When it comes to observation, there are varying degrees to which the researcher can become integrated into the scene. At one end of the spectrum the researcher may become an observer-participant. In this role, one attempts to be as unobtrusive as possible, quietly observing events from a discreet, yet strategic, position. At the other end of the spectrum is the participant-observer. In this situation, the researcher is actively involved in the events observed (e.g., a researcher who goes through the training to be a machine operator in an industrial environment).

There are pros and cons associated with each type of role. While being fully integrated into the action provides a researcher with firsthand experience of events, taking good notes in this context is difficult at best. A great deal of energy is spent trying to fit in rather than on attempting to make sense of the events in the context of the research objectives. In such cases, one must rely on memory of the events when writing up field notes after the fact. Taking a more observational role affords a wider perspective on events and the time to record and reflect on events as they unfold. On the downside, it precludes the opportunity to experience the activity firsthand. In many research situations, the ethnographer's position moves between these two extremes, sometimes occupying a hybrid position of both partial participant and outside observer.

45.7.4.3 Structuring Field Observations

Before setting out to observe, decisions need to be made about what, where, and when to observe (Whiting and Whiting 1970). One might decide to observe individuals as they go about their work and daily routines (person focused), a technique sometimes referred to as "shadowing" (Wasson 2000). The researcher might also decide to focus on a specific event, such as a meeting or software education class (event focused), or observe the activities that occur over time in a given area, like an office or store (place focused). One can even shift the subject of observation to an artifact, such as a document, and record its transformation as it moves from person to person or along a development path (object focused).

45.7.4.4 Video Recording

Given the complexity of human activities it is impossible to notice and record in real time everything of interest to the researcher. This is one reason video cameras have become increasingly popular in fieldwork. Video records can be used as a reference to supplement field notes. The ethnographer also has the advantage of being able to watch events multiple times and change levels of analysis or observational focus with subsequent viewings (e.g., interaction between people vs. the movement of one individual in and out of a scene).

Video recording also allows people not primarily involved in the fieldwork to participate in the analysis and opens up the range of perspectives that can be bought to bear on the analysis (e.g., Blomberg and Trigg [2000] used video collections in interactions with product developers; also see Brun-Cotton and Wall [1995]; Karasti [2001]; Suchman and Trigg [1991]).

Video cameras can also be used to record events in the absence of the researcher. Not only does this free the researcher to be involved in other activities, but the camera also can be a silent presence* in situations where an outsider (even a well-trained participant observer) would be seen as intrusive (e.g., child birth, counselor-student interactions, board room deliberations, etc.). Video recording however requires devoting time later to review video records and incorporate relevant information into the analysis.†

* However, the expressed permission of the participants in the interaction is needed in these cases as well.

† A variety of software applications now exist that can help the researcher manage and analyze recorded on video. Caveat, for example, allows the researcher to select and annotate images/events of particular interest. A more sophisticated (though less user friendly) program is observational coding system that provides for a more quantitative analysis.

45.7.5 Interviewing

Interviewing is a central tool of ethnographic research (Gubrium and Holstein 2002). Conducted and interpreted in light of the potential differences between what people say and do, interviews are critical in developing understandings of members' perspectives. Interviews can be placed on a continuum from unstructured to structured, with at one extreme the casual conversation and at the other a formal structured interview.

Ethnographic interviews are most often open-ended, particularly during the early stages of fieldwork when the ethnographer is just beginning to get a perspective on the activities and people studied. The more unstructured format gives the researcher the freedom to alter the line of questioning as the interview unfolds. The researcher essentially is learning what questions are important to ask. Unstructured, however, does not mean haphazard or lacking purpose. The researcher will know the research objectives and the topics to be explored when entering the field, and will usually have an interview protocol to serve as a (flexible) guide for the interview. While the protocol provides a basic framework for an unstructured interview, the participant plays a major role in the direction the interview takes. As Bernard (1995) wrote, the idea is to "get an informant on to a topic of interest and get out of the way." When the interview moves to a topic of particular interest, the researcher can then probe deeper to elicit more details. Indeed, interviewing is something of an art, and one of the key skills an ethnographer learns is the art of "interrupting gracefully" (Whyte 1960).

In an open-ended interview it is important to avoid using an interrogation style of questioning (e.g., "yes or no" questions) which is designed to uncover the "facts." This defeats the purpose of keeping the interview open to allow for a wide range of responses and for the participant to express his experiences, in his own way, in his own words. Using too structured a format constrains the range of possible answers, increases the chances of missing critical pieces of information, and increases the risk that discoveries will be limited by the ethnographers' preexisting concepts, assumptions, and hypotheses. It is critical to provide opportunities for participants to convey their stories and perspectives in their own way and for the researcher to be surprised by what is said.

As a project progresses and patterns begin to emerge, interviews can become more structured and the line of questioning less broad. The researcher begins to narrow in on topics that are particularly informative and relevant to the research objectives. Questions become more focused and specific as answers to previous questions guide the follow-up questioning.

Once the range of responses is known and themes begin to emerge, the researcher may want to structure interviews further. A host of structured techniques exist. Some are designed to identify the ways people organize information within a specified domain, such as free listing, card sorts, triad's tests, and paired comparisons (Romney, Batchelder, and Weller 1986; Weller and Romney 1988). Other techniques, such as questionnaires and surveys,* are used to assess variations between two or more groups or to establish the representativeness of the findings for a larger population. The main idea behind these techniques is to keep the form and content of the questions consistent for each respondent, thus allowing for differences among the sample population to be ascertained. Conducting structured interviews at the end of an ethnographic study has the advantage of allowing the question structure and language to reflect the way participants talk about and organize experiences, thus increasing the validity of the survey findings.

45.7.5.1 The Interview as a Communicative Event

The interview has become somewhat ubiquitous in western societies and is viewed as a reliable means of acquiring information of all kinds (e.g., attitudes toward tax increases, the value placed on education, preferences for certain products, basic demographic data, etc.). However, as Briggs (1983) points out, what is said in an interview should not be thought of as "a reflection of what is 'out there'" but instead must be viewed "as an interpretation which is jointly produced by the interviewer and respondent" (p. 3). This view compels us to regard the interview as a communicative event in which the structure and context of the interaction conditions what the researcher learns. This is no less the case in highly structured interviews (see Jordan and Suchman [1990] and Moore [2004] for a critical analysis of the ecological validity of survey research). Briggs recommends that we adopt a wider range of communicative styles in our interactions with study participants, particularly styles that are indigenous to the study population.

45.7.5.2 Interviewing Rules of Thumb

While there are no hard and fast rules for interviewing, a few general guidelines will help facilitate the interview process and increase the chances of obtaining useful information. The following are some points to remember:

- Interview people in everyday, familiar settings. Not only does this make the participants more comfortable, it allows them to reference artifacts in the environment that play an integral part in their activities. Moreover, a familiar environment is full of perceptual cues that can help jog the not-so-perfect human memory.
- Establish and maintain good rapport with participants, even if it slows the interview process.
- Do not underestimate the value of casual conversation. Some of the most insightful information comes from informal conversations when social barriers are lowered.
- Assume the respondent is the expert and the researcher the apprentice. This not only shows the

* A good introductory book on surveys is How to Conduct Your Own Survey (Salant and Dillman 1994). Readers interested in a more advanced treatment of the subject are referred to Babbie (1990).

respondent respect, but also gives them confidence and facilitates conversation. Even if the interviewer happens to be more knowledgeable on a particular topic, the goal of an ethnographic interview is to understand the respondent's perspective.

- Use lack of knowledge as a discovery tool. Respondents will always know more about their own experiences than the interviewer. In this context, do not interrupt unnecessarily, complete a respondent's sentences, or answer the questions. Again, the idea is to learn about the respondent's point of view, not the researcher's. In this context, the researcher's "inevitable ignorance" about the experiences of another person can be a powerful tool.
- When conducting an open-ended interview, avoid asking "yes or no" questions. Responses to these questions provide less information than questions beginning with "what" or "how."
- Be flexible enough to adapt the line of questioning when necessary. Human experiences are complex and full of surprises.

45.7.6 Connections between Observation and Interviews

As noted earlier, one of the defining qualities of ethnography is its emphasis on holism. To obtain this holistic view, combining different sources of data is useful (Agar 1996). Observation alone is seldom enough to adequately address research objectives. As such, observation is invariably coupled with interviewing. Interviews can extend and deepen one's understanding of what has already been observed. Similarly, interviews can be conducted prior to observing, giving the researcher a better idea about what is most appropriate to observe.

Interviews can also be conducted in the context of ongoing activities, sometimes referred to as "contextual" or "in situ" interviewing. Instead of setting aside a specific time and place for an interview, the researcher creates an opportunity to ask questions as participants go about their daily activities. The strategy can be extremely useful in getting answers to questions that are prompted by observation of ongoing activities.

45.7.7 Self-Reporting Techniques

In cases where the domain of interest transpires over a long period, or in which direct observation is not practically feasible, self-reporting techniques can be very valuable. This methodology is especially good at revealing patterns in behavior or obtaining data that is otherwise inaccessible (Whyte 1984). A number of self-reporting techniques exist which vary in terms of form, focus, structure, and mechanism of self-reporting. Common techniques range from simple written diaries to visual storybooks, and more recently to Internet-based blogs.

45.7.7.1 Diaries

Traditional diaries consist of written records, which might include personal thoughts or descriptions of specific behaviors or accounts of events in which an individual participates (Zimmerman and Wieder 1977; Carter and Mankoff 2005). The focus, format, and degree of structure of diaries used in ethnographic research vary depending upon the research objectives, ranging from structured activity logs which invite participants to capture and describe specific aspects of their experiences for each entry, to relatively unstructured forms in which diarists are provided only with general instructions. Study participants might be asked to keep diaries regarding the specific contexts, foci, modalities, and outcomes of their interactions or they might simply be asked to describe their experiences over time while using a specific product.

Diaries are obviously not a substitute for direct observation. However, they are valuable tools for ethnographers, expanding opportunities to learn about behaviors that cannot be observed because of practical constraints and limitations on time and resources (Gillham 2005).

How diaries are analyzed depends on the research objectives and resource constraints. If time permits, follow-up discussions with participants to clarify points or gain a deeper understanding of the meaning behind the words can be useful. The texts can also be coded for themes, key words, or phrases and patterns examined across individuals or between groups.*

45.7.7.2 Visual Stories

Visual stories are essentially pictorial diaries that employ images in addition to text in order to document experiences. They can be particularly valuable when working with language limited participants, such as children, or in situations where words alone are inadequate to capture the essence of the subject (Johnson et al. 1997). Much like more traditional text-based diaries, visual diaries can be employed and structured in any of a number of ways. Wasson (2000), for example, described giving participants a written guide directing them to take photographs of their interaction with a product under study. They were then asked to organize the developed photos into a story that made sense to them, and researchers conducted follow-up interviews over the telephone.

A more open-ended framework can also be informative. Interested in cultural differences between Italian and American fishermen, Johnson and Griffith (1998) instructed participants from both groups to take photographs of whatever they wanted. After developing the film, Johnson coded the pictures based on their content and found significant thematic differences between the groups, which added to his understanding of differences in cultural values of the two groups of fishermen.

A more recent derivation of the visual story utilizes a video camera which allows the participant to provide a running

* With varying degrees of success, text analysis software has been used to help with large data sets. Some noteworthy programs are Ethnograph, NUD*IST, E-Z-Text, and NVivo.

narrative alongside the visual content. Being able to experience the two sources of information simultaneously provides the researcher with a rich record of an activity. Blomberg, Suchman, and Trigg (1996) used a video-story approach in their study of the document practices of lawyers. They set up a stationary video camera in the law office of a study participant and asked him to turn on the camera whenever he had occasion to retrieve documents from his file cabinet. The running narration recorded on videotape provided insights into the everyday use of the file cabinet that helped inform the design of an electronic file cabinet. The pervasiveness of mobile phones with built-in cameras has enabled the use of the visual diary technique. Researchers are able to set up urls where participants can upload their photos or video recordings with commentary as they document selected aspects of their lives (Palen and Salzman 2002).

45.7.7.3 "Blogs" and "Tweets"

Online tools for self-reporting, communication, and social networking have continued to evolve rapidly. The concept of weblogs or "blogs"—in which a website is used to post online entries that may include textual narratives, digital photos, or digital video or audio (Nardi, Schiano, and Gumbrecht 2004) has more recently been followed by the broad adoption of digital tools for "microblogging." The latter is best exemplified by the popular service provided by Twitter (founded in 2007), which enables someone to publish and disseminate short text "microblogs" or "tweets"—text messages of up to 140 characters—for others to read or "follow." Although not developed specifically to support ethnographic inquiries, blogs and "microblogs" can be potentially very valuable research tools. Blogs may be particularly useful as a way for participants to self-report their use of online tools in the context of their online activities. Blogs also enable researchers to review participant posts as they occur as well as to engage in asynchronous online exchanges and dialogues with participant "bloggers." These interactions might be viewed as virtual analogs to the questioning that occurs during shadowing or on site observation. Indeed, as blogs increasingly are used in ethnographic research (e.g. Berry and Hamilton 2006), they may blur the boundary between self-documentation and interviews, resulting in a blend of online self-reporting and intermittent online "conversations" via threaded participant and researcher posts.

In addition, some ethnographers have begun to experiment with posting their research notes via blogs, enabling research and design team members to review and comment asynchronously and in near real time. By making observational and interpretive notes more readily visible to teams, the ethnographer may promote dialogues which can inform further observations as well as accelerate and heighten impact of research on design.

Microblogs or "tweets" can provide near real-time snippets from the stream of experience when direct observational shadowing is not possible or practical. Indeed, although not specifically directed at ethnographers, Twitter actually provides tips ("twitip") about why and how to do research with Twitter (http://www.twitip.com/twitter-for-research-why-and-how-to-do-it-including-case-studies/). While blogs and tweets can be done quickly, relatively easily, and while participants are mobile, as with any self-reporting methodology, there is always the possibility that the activity of self-reporting may alter the phenomena being studied.

45.7.8 Remote "Virtual" Observation: Digital Ethnography

Continuing technological developments—in video, audio, wireless, network applications, global positioning system (GPS) tracking capabilities, and pervasive computing—have created new opportunities to "observe" and collect rich and dynamic information across geographies in real time as well as asynchronously. These technologies increasingly enable ethnographers to "virtually" observe in a wide variety of contexts. Using digital video and audio, people's behaviors can be tracked and analyzed as they interact with computer supported products and Internet-based networks.* Indeed, these technologies (along with the use of other digital tools such as blogs and microblogs) enable what some have begun to refer to as "digital ethnography" (Masten and Plowman 2003; Murthy 2008; Mason and Dicks 1999; or even "netnography" by Xun and Reynolds [2010]).

The pervasiveness of the "webcam" is perhaps the simplest illustration of how technology has expanded the observational capabilities of ethnographers. Internet-enabled digital video cameras can stream video in real time and can be remotely controlled. This digital video and audio can be viewed by multiple people across geographies either in real time or by accessing video archives. Such techniques and information sources can be particularly useful for geographically distributed research and design teams or where the activities of interest are widely distributed making direct observation difficult.

In addition, computer and online sensing, tracking, and analytic technologies that monitor, gather, collect, and integrate information on computer mediated activities can be a useful source of information for ethnographers. Although early tracking and analytic technologies required complex sifting and analysis of massive amounts of data to find meaningful nuggets, more recent tools enable sophisticated tracking of individual paths and activities as well as the ability to model online behavior. For example, scenario-based behavioral models (e.g., of online shopping, exploratory behavior, task completion, etc.) which define hypothesized patterns or sequences ("funnels") of online behavior can be used as an analytic lens to understand individual or group online behaviors. To date, these tools have been used primarily to measure aggregate completion of online tasks (e.g., online shopping, self-service) and to identify obstacles to user success

* The ability to virtually observe and track behaviors presents many ethical issues that cannot and should not be ignored. It is critical that ethnographers establish guidelines and protections if they engage in electronic, digitally enabled observations.

(e.g., usability issues). However, over time and in conjunction with other sources of data and information they may become useful tools for ethnographers interested in patterns of online behavior and technology adoption. This may become particularly important as ethnographers attempt to understand the formation and interactions of distributed virtual communities (e.g., Rheingold 2000; Wilson and Peterson 2002).

The potential for using (and misusing) these sources of information will likely increase as pervasive computing increasingly enables the identification of (and response to) individuals across multiple physical and digital environments and the tracking of their activities. The collection and use of digitally enabled behavioral observations obviously needs to be carefully constrained by ethical considerations, particularly the respect for privacy and informed consent. In addition, as with any behavioral observation, it is critical to understand the context in order to interpret the meaning and significance of the behavior. In this respect, tracking computer-mediated behaviors by itself is insufficient and may simply result in the collection of massive amounts of relatively meaningless data. However, if used in conjunction with other sources of information (e.g., self-reports that illuminate peoples' intentions and meanings), patterns in digital behavior can illuminate aspects of behavior that are difficult or impossible for a human researcher to observe.

For example, it has been increasingly common for teams designing online services and tools to examine individual and aggregate patterns of online behavior (as reflected in web server logs or "client side" logs that are generated as a function of what users do online) to both identify usability issues as well as to examine patterns of technology, product, and service adoption over time (Kantner 2001).

45.7.9 Artifact Analysis

Ethnographers have long had an interest in the material world of the people they study (Appadurai 1988). The artifacts people make and use can tell us a great deal about how people live their lives.* Artifact analysis can be an important part of contemporary ethnographic studies (e.g., Rathje and Murphy 1991). For example, conducting an artifact analysis of the stuff on people's desks can say a great deal about the people's work practices (Malone 1983). Similarly, studying the contents of an automobile's "glove box" can tell a great deal about how the car is used. Depending on the kinds of research questions asked, it may be useful to include the collection and analysis of specific artifacts.

45.7.10 Recordkeeping

Although the authority of the ethnographic voice derives in part from the fact that the ethnographer is present and witness to events of interest, the ethnographer should not rely exclusively on experiential memory of these events. In all ethnographic research it is essential to keep good records. Field notes should be taken either during or soon after observing or interviewing. The specific nature of the notes will depend on the research questions addressed, the research methods used, and whether audio or video records supplement note taking. Field notes should at least include the date and time when the event or interview took place, the location, and who was present. Beyond that, notes can vary widely, but it is often useful to indicate differences between descriptions of what is observed, verbatim records of what is said, personal interpretations or reflections, and systematic indications of the flow of observed events and activities. When working with a team of researchers, field notes need to be understandable to other team members. This is often a good standard for the specificity of field notes even when working alone. If such a standard is maintained, it will be more likely that the notes will be useful to the researcher months and even years later, in the event reanalysis or a comparative study is undertaken.

45.8 QUALITATIVE AND QUANTITATIVE DATA

In Section 45.7.6, we touched upon the complementary nature of observational and interview techniques and the benefit of combining these two approaches. Triangulation of data can serve to connect quantitative and qualitative data as well. Sometimes, prior to the start of a project the only data available is quantitative, sometimes in the form of survey data focused on population characteristics. Qualitative data derived from ethnographic research can complement quantitative research by providing a meaningful context for interpreting the quantitative results. Qualitative techniques allow researchers to dig deeper after a survey has been tabulated, and aid in interpreting and explaining trends that the quantitative data might reveal (Guest 2000). In addition, qualitative data can inform the content and language of more structured questions, thus making them more meaningful and relevant to the participants.

45.9 ETHNOGRAPHY IN A GLOBAL CONTEXT

While ethnography has its roots in the study of small-scale, non-Western societies, the application of ethnography in the design of products and services until very recently has focused primarily on groups and individuals located in the developed regions of the world (e.g., North America and Europe). Two recent developments have led to a shift in the center of design activity. One is the emergence of the economies of less-developed countries, particularly Brazil, Russia, India, and China, the so-called BRIC countries, which are rapidly becoming major markets for products and services. An interest in serving these growing markets has led some firms to invest in designing products and services specifically for them by directly engaging designers, developers, and potential users from these developing regions (e.g., Bell 2004; Foucault, Russell, and Bell 2004). The second development is the rapid increase in the use of Internet-enabled information

* Archaeologists rely almost exclusively on the artifacts that remain in archaeological sites for their interpretations of the behavior and social organization of past human societies.

technologies that connect workers, consumers, citizens, and organizations distributed around the world including developing regions.

45.9.1 Globally Distributed Interactions

The methods and techniques of ethnographic research must contend with the increasing number interactions both at work and in domestic spheres that take place "virtually" between people separated in space and time. This is a challenge for ethnographic techniques that were developed to study communities of people who interact face-to-face. Ethnographic techniques and approaches must be adapted both practically and analytically to this new context where many more interactions are mediated by information technologies (e.g., instant messaging, blogs, twitter, SMS, e-mail, telephone, web conferencing, shared digital workspaces, and repositories) that transform traditional notions of place, community, and real-time interaction. In many enterprises, work teams are made up of people who are not co-located, many of whom are highly mobile in their work activities, requiring interactions to take place through conference calls, instant messaging, SMS, e-mail, and even microblogging (Zhao and Rosson 2009). Furthermore, in some regions of the world, people travel significant distances for jobs and other opportunities. In these cases, interactions with friends and family, as well as with others living away from their native communities, are enabled by communication technologies (Horst and Miller 2005; Green, Harvey, and Knox 2005). Various strategies have been developed to study distributed, multisited groups including team ethnography (placing researchers in multiple locations), perspectival ethnography (focusing on the view from one of the local sites), and virtual observations (observing digitally mediated interactions).

45.9.2 Multisited Ethnography

The challenges of studying ethnographically an increasingly interconnected and globally distributed world became an important topic for anthropologists in the mid-1990s with the publication of an article by Marcus (1995) that raised the question of whether and in what ways ethnography, with its traditional reliance on the "field site" was well suited for the study of the contemporary experiences of people around the world. Taking a multisited approach was viewed as a way to apply an ethnographic perspective to theoretical and practical concerns in the study of migration, Diasporas, technologically interconnected (virtual) communities, and globalization more generally. Multisited ethnography addresses strategies for studying geographically distributed activities and groups (Coleman and von Hellermann 2011; Falzon 2009; Hannerz 2003).

Instead of defining the field site as a single location the emphasis is on capturing the connections between people, places and things regardless of their geographic proximity. This then raises the question of how to circumscribe the study in the absence of having a single field site. However, local alone rarely delimits an ethnographic study relying more often on the research questions being addressed to define what is in and out of focus. On this view the bounds of the study is always constructed and cannot be given by place alone.

That said there is still the practical issue of how to limit the many possible physical sites in order to devote enough time to any one of them. One strategy has been to pick a "focal" site and then move beyond the focal site as research dictates given the resources available. For example, while a particular call center might be deemed the focal site for the research, other sites such as workers' local domestic residences or the more distant communities of their origin might also be included in the study.

45.9.3 Shifting Center of Design Activity

As new markets open up around the globe, many businesses and organizations see opportunities to create products and services specifically for these markets recognizing that the products and services suited for the developed West may not be appropriate for these other regions. As such these firms may establish design initiatives focused on and located in countries like India and China. In some respects ethnography has come full circle with the application to design, contributing to understandings of the contexts of people living in culturally and linguistically diverse settings (the sites in which ethnographic practice first developed). More than ever ethnographic principles and practices are applicable and necessary as the center of design activity moves outside the developed West, in particular the principle of members point of view and the focus on "what is" as a resource for innovation and design.

45.10 DESIGNING WHAT?

The application of ethnography to support a design agenda was directed initially toward informing the design of technologies, tools, and products. However, attention has expanded to include the use of ethnography to inform the design of experiences (Pine and Gilmore 1999), services (Kieliszewski, Bailey, and Blomberg 2010; Kimbell 2009; Mager and King 2009; Thomke 2003), organizational processes, and business strategies and models. In addition, educational institutions and programs (e.g. the "d-school" at Stanford University; Design Ethnography program at University of Dundee, Applied Anthropology programs at California State University at San José and North Texas State) are dedicated to teaching design thinking to address a myriad of problems beyond the design of products. Ethnographers are now involved in projects and contexts that span a range of problems including the design of the next e-mail application, business models to reach small and medium businesses with IT services, customer services for retail banking, integrated health care services, and Internet delivered social networking services.

45.10.1 Products

The application of ethnography to product design has received the most attention in the literature partly because many of the pioneers in the field worked in corporate research organizations of major technology companies (e.g., Xerox, Apple, and HP). In addition, early commercial applications of ethnographically-informed design often focused on the design of consumer products, from cleaning products to automobiles to toys (Elab, Doblin group, Sonic Rim). It is not surprising therefore that many view product design, whether high-tech products like personal digital assistants and online calendar applications or everyday consumer products like breakfast cereals or cold remedies, as the primary application of ethnographic research (Squires and Byrne 2002).

45.10.2 Experiences

The publication of the Experience Economy (Pine and Gilmore 1999) marked a shift in design focus to include the experiences that products and other artifacts enabled. Pine and Gilmore argued that the real challenge for businesses is creating engaging experiences for both consumers and corporate customers. The admonishment by a number of business gurus to pay more attention to the customer in the design of products also contributed to this expanded focus. Customers, it turned out, cared less about the products themselves and more about what the products enabled them to do or experience. Businesses became concerned with delivering quality experiences in which the products took on more of a supporting role. The canonical example often cited for this shift to an experience economy is Starbucks, where what is being sold is not simply a cup of coffee, but the experience of buying and consuming the coffee at Starbucks, including the elaborate choices available, the wireless access provided in the stores, the exclusive access to trendy music, and so on. In many commercial contexts user experience design has become the new moniker for the application user-centered design approaches, including ethnographically-informed design, to the development of new products and services.

45.10.3 Services

The service sector has come to dominate much of the world economy and increasingly new services are the site of significant change in the way we work and play (e.g., online dating services, GPS tracking services, business process outsourcing services). Many innovative services are enabled by new technologies that provide the platforms* on which new service relationships are built. However, the service is marketed and not the technology that facilitates its delivery. Ethnographically-informed design strategies are now being applied to service design (Kimbell and Seidel 2008; Jones and Samalionis 2008; Mager and King 2009; Thomke 2003). In addition, recent advances in Web 2.0 technologies have created opportunities for the development of a wide range of services, including public services to improve government, health care and community services, and commercial services to enable firms to connect with their customers (c.f. Scola-Streckenbach 2008; Dittrich et al. 2003).

Kimbell (2009) argues that service design developed in conjunction with changes in design practice brought about by the widespread use of networked media technologies. The "outputs" of designs were expanded beyond stand alone technologies to include the "arrangement of interfaces to distributed devices." And through these interfaces new services were being delivered. On this view the focus of design was on the services delivered and less on the individual devices that enabled the delivery. Many small firms and in-house research and design departments now offer service design along with product and experience design.

45.10.4 Organizational Processes

Workflow systems have become ubiquitous within many organizations, orchestrating everything from employee travel-reimbursement processes to customer online-purchasing procedures. With this comes the opportunity to inform the design of these technology-enabled organizational processes through the study of existing work practices and processes. Here again the design focus is not so much on the underlying technologies (e.g., SAP, Siebel) that manage the workflow, but on the processes themselves. This is not to say that these studies will have no impact on the underlying technologies—for example, making them more flexible or end user configurable. But the design focus is on the workflow requirements, how people will interact with these systems and will be supported in executing processes (Bowers, Button, and Sharrock 1995; Dourish 2001; Randall, Rouncefield, and Hughes 1995).

45.10.5 Business Strategies and Models

Ethnographic research is also playing a role in the design of business strategies and models. Organizations are realizing that their competitive advantage is only partly related to the quality of their products and services. Equally important are the business strategies, including channels to the market, relationships with business partners, and the composition of employees. Many new business models have emerged in the last decade that capture new revenue streams such as advertising (e.g., Google, Yahoo!), selling software as a service (e.g., salesforce.com), and facilitating networks of sellers and buyers or customers and providers (e.g., eBay, regional IT distributors). Ethnographic research is contributing to the design of these new business models.

45.11 Making Ethnography Matter: Communicating and Applying Ethnographic Insights to Design

This section outlines some of the ways in which the insights derived from ethnographic work can be represented and communicated in order to effectively inspire and guide the design of products and services. These ways of representing and communicating what is learned are intended as examples

* See, for example, service-oriented architectures.

of how ethnographic work can be made relevant for design. However, before we outline some of these representational forms and practices we should consider the possible purposes of our representational activities.

45.12 ENHANCING THE WORKING MODELS OF DEVELOPERS

In order to design a product or service for people, designers must have at least an implicit working view of the people who will interact with the system. Such working frameworks and perspectives may include assumptions about a range of essential characteristics of the people who will engage with the product or service and the contexts in which they will do so (Newman 1998). Indeed, some would argue that successful design requires a high degree of "empathy" with the target population (e.g., Leonard and Rayport 1997; Koskinen, Battarbee, and Mattelmäki 2005). Implicit and/or explicit assumptions or knowledge about "users" may be formed through some combination of direct experience (e.g., interacting with and/or observing people in the target population) and secondary learning (talking with others about the target group, viewing videotapes of target activities, reading, analogy to other directly experienced groups, etc.). However formed, the working "models" of designers/developers may be of varying levels of complexity, robustness, coherence, consistency, and viability. The broad, deep, and contextualized understanding provided by ethnographic research can enrich the design team's implicit working models.

45.13 SUPPORTING INNOVATION

The design of products and services for people obviously poses a range of potential creative challenges at varying levels of complexity. What problems should be solved? What should be built? What kinds of experiences should be supported or enabled? What features and functions would be useful, compelling, and satisfying for a particular group of people in a particular domain or context? How can current or emerging technological capabilities be used to enhance a particular group's experiences, or to solve a particular problem? Even if there are clear parameters defining the functionality that will be built (e.g., a set of "requirements"), design teams must still generate a compelling, easy to use, useful, and satisfying way of delivering that functionality. By providing an understanding of the human domain (patterns of relationship, systems of meaning, organizational structure, guiding principles or rules, etc.), ethnography can promote creativity that matters (Robinson and Hackett 1997)—relevant innovations that create new, realizable opportunities.

45.14 EVALUATING AND PRIORITIZING IDEAS

Design teams not only face the challenge of generating innovative ideas and concepts, but also the equally important task of evaluating and prioritizing ideas and options that arise from various sources (e.g., business stakeholders, end users, development teams). Although there are obviously many evaluative methods (e.g., scenario-based user testing, etc.), models derived from ethnographic research and analysis (e.g., scenarios, mental models, process models, personas, etc.) can provide a critical lens through which development teams can evaluate and prioritize ideas based on how they may fit into or change people's experiences. The need for evaluation and prioritization may occur at various points throughout the development process, ranging from decisions about features and functions, broad directions for design concepts, and so forth.

45.15 SHARED REFERENCE POINTS

The learning derived from ethnographic analysis, particularly when represented as explicit representations and models, can serve as an experiential guidepost for individual designers and design teams throughout the development process. Even though these representations do not prescribe or specify what should be done, they can aid developers by focusing attention on essential aspects of an experience, highlighting variations in the experiences, and limiting exploration of experiential "dead ends." In other words, they can provide a general structure and direction within which a team can develop a shared understanding and focus its creative energies.

45.16 INFORMING USER ADOPTION STRATEGIES AND PLANS

Ethnographic insights not only inform the design of products and services, but they also guide the generation of effective strategies for promoting adoption of solutions. Understanding the current state, how people operate and view their experiences today can enable the identification of experiential and social barriers inhibiting adoption. This understanding also can point to the levers or factors that can be used to overcome barriers to change and accelerate adoption. Particular designs will invite or require changes in user behavior and experience which can be better anticipated if the current state is well understood. Ethnographic understanding enables designers and change agents to move beyond the general factors outlined in some of the major models of user adoption (e.g., Rogers 2003 diffusion model) to specific insights and contextual factors that are meaningful and important to specific sets of users. For example, cross-cultural ethnographic research on current work practices conducted by the second author in a large global enterprise prior to the rollout of a new business process and an associated web-based tool, highlighted the differential impact that the solution would have on users in varied geo-cultural settings. In this case, users in Japan would be asked to shift from a highly cumbersome manual process, primitive offline tools and artifacts, and an unreliable and confusing collaborative process to what for them would be a highly streamlined, labor saving set of tools and work practices. In contrast, users in key North American settings would be asked to shift from a highly automated, simple, and familiar process to one that involved learning a new tool, and a new process that added extra work steps. General communications to promoting the new tools and processes as "easy to use" and "labor saving," would clearly have fallen on deaf ears in North America while potentially resonating with users in Japan.

45.17 REPRESENTATIONS AND MODELS

Whether the focus is on designing products, experiences, services, processes, or business strategies, the researcher must find ways to ensure that ethnographically derived insights effectively inform design innovations and decisions. Researchers can help make connections between ethnography and design in many ways. At the most basic level, this is achieved through active engagement, integration, and collaboration of researchers and designers.* Subsequent to conducting ethnographic inquiries, researchers can engage with design teams by acting as user proxies (e.g., helping to formulate and/or review design concepts in scenario-based reviews, providing feedback regarding relevant user expectations and behaviors as they relate to design concepts and decisions, etc.). Conversely, the active and direct involvement of designers in key elements of ethnographic fieldwork (e.g., participating in observations and interviews, collaborative analysis sessions, reviewing video and audio recordings and user artifacts, etc.) can enrich their understanding of the people who will interact with and use the solutions they design.

Although these forms of engagement are valuable, they limit the ability of teams to take full advantage of ethnographically derived understandings. They are restricted in the impact to the scope of the direct interactions between ethnographers and designers. This can be particularly limiting when designing multifaceted solutions, working with large and/or distributed design and development teams.

45.17.1 The Value of Representations and Models

To increase the impact of ethnographic research, explicit representations or models can be created which distill and communicate essential insights about people's experiences in forms that can be applied to design problems and decisions. Although the definition of model can be the subject of debate (as can the distinction between representation and model), for our purposes we are using the term to refer to explicit, simplified representations of how people organize and construct experiences and operate in relevant domains. The important point here is that well-constructed representations which communicate effectively can help connect everyday patterns of activity and experience with design solutions. More specifically, representations and models are tools that can serve a number of purposes including enhancing the working models of designers/developers, supporting innovation and creativity, evaluating and prioritizing ideas and concepts, and providing shared reference points for design teams.

45.17.2 Types of Representations and Models

Representations and models can vary, ranging from personas and scenarios to more abstract mental models. The number, type, and form of models vary as a function of what is being designed, the audience, and the constraints on the design process (e.g., Chapters 42 and 43). For example, teams designing organizational tools may find it useful to model work environments and detailed task sequences; teams designing learning tools and programs may want to represent particular skill domains, as well as learning processes.

Practitioners have developed a variety of representations and models to inform the design-and-development process. For example, Beyer and Holtzblatt (1998) described a set of five work models (flow model, cultural model, sequence model, physical model, and artifact model) to reflect different aspects of a work domain. Pruitt and Grudin (2003) articulated the value (and risks) of personas to inform the design process, while Carroll (2000) described the value of scenarios.

The varying scope, form, complexity, and function of different types of models are illustrated in following examples.

45.17.2.1 Experience Models

The model presented in Figure 45.2, is one of several developed in the context of ethnographic research and analysis for a financial services company serving individual investors. This company aimed to develop web applications that would facilitate customers' active engagement in the investment process with particular financial instruments. The model was intended to articulate and visualize a financial development process as well as the varied meanings of "money." This particular model highlighted the role of "practice" in developing the confidence and knowledge to become engaged in the investment process, and the iterative/recurrent nature of the process, as people learned to deal with new financial instruments and domains (e.g., securities, bonds, options, etc.). Moreover, it illustrated the distinctions that people make between "real," "play," and "foundational" money and the relationship between these categories, investment behavior, and financial development. To oversimplify a bit, people are more fully engaged and active in the investment process when they view the assets/investments as "real" (e.g., money that is used to address their current and emerging needs, pay bills, etc.) rather than as "play" (e.g., stock options that are perceived as intangible and somewhat imaginary) or "foundational" (e.g., savings for the future that are left "untouched"). As people have an opportunity to "practice" and develop their knowledge, they may move from construing a particular financial instrument or activity as "play" to "real." These notions suggested that web applications in this domain should not be focused on simply providing a wealth of financial information or a plethora of tools. Instead, these patterns helped to foster the generation of numerous ideas about ways to engage people in playful learning in the financial domain, with the aim of facilitating the financial development process.

* As noted earlier, the ethnographer should develop an understanding of the types of design decisions that the design team will need to make and a sense of what they need to know to inform those decisions.

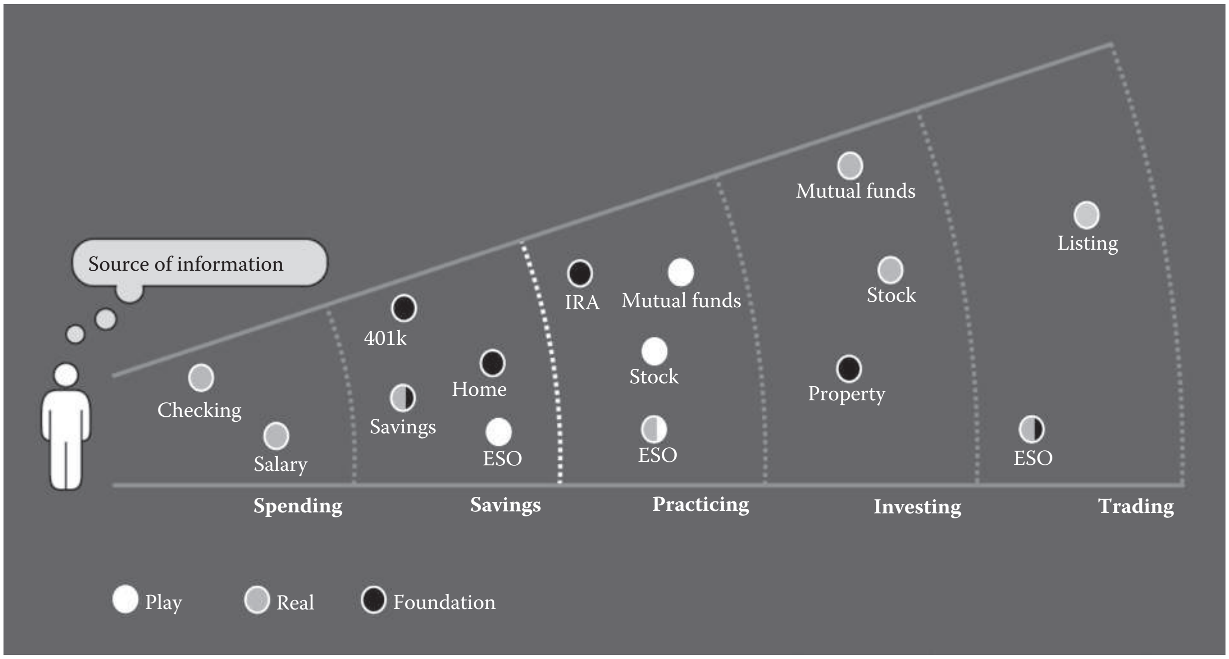


FIGURE 45.2 Experience model of financial development zones.

45.17.2.2 Process Models

Process models attempt to represent how a dynamic experience "works" and/or unfolds over time. They can range in focus from relatively circumscribed task-flow models that outline how an individual completes a specific task, to broader characterizations of more holistic change processes (e.g., health care behavior change, technology adoption, etc.). For example, a health services company aimed to develop an "electronic medical record system" (combining client server applications with web-based "portals"). This system would, among other things, increase the efficiency and effectiveness of their medical practice, enable patients to view their health records online, and ultimately empower patients and foster a proactive approach to wellness and health care (both by clinicians and patients).

At the outset of the engagement, the health services company had generated a rather long requirements list (several hundred features and functions) and a particular view of the structure and function of the web components of the system. It was clear that the budget for this initiative was not sufficient to build a system that met all of the initial "requirements." Perhaps more importantly, it was unclear which components would ultimately add the most value for the various stakeholders (clinicians, patients, the business owners, etc.). Ethnographic research examining the experiences of and relationships between clinicians and patients in context (in clinic settings and in homes) provided the means of prioritizing and evaluating potential features, functions, and design concepts.

Experience models of varying levels of complexity regarding the health management process were developed. For example, one of the simpler models (see Figure 45.3) described how individuals, in the process of adopting an active/proactive stance in relation to health issues, move through varying "stages of readiness." A more comprehensive, integrative model highlighted the ways in which various factors interact in influencing a person to take action in addressing a health issue and mapped the role of various health care-related activities (e.g., monitoring, motivating, learning, sharing, building rapport) in various stages of readiness. The combination of these models enabled the team to identify the most important opportunities for facilitating progression towards a proactive orientation to health, and provided guidance in identifying ways to provide messages and experiences tailored to a person's stage and readiness.

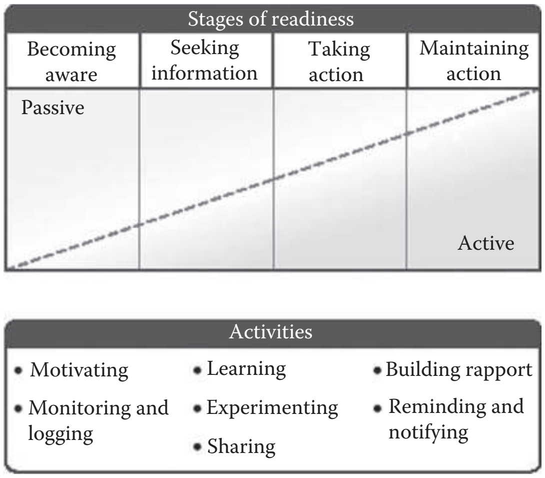


FIGURE 45.3 Stages of readiness model.

45.17.2.3 Personas

One of the primary challenges in developing interactive systems is to design them so that they meet the needs of varying users, who may play different roles, engage in varied tasks, have different motivations and strategies, and so forth. Profiles or personas are abstract representations of the users of a solution (Pruitt and Grudin 2003; Pruitt and Adlin 2006) that may be informed through ethnographic studies. Personas can help development teams understand and anticipate how certain types of people may experience and interact with technology solutions or services. For example, Figure 45.4 shows a simple persona developed to guide the design of interactive tools promoting the adoption of various financial and health benefit programs in a large enterprise. Note that the persona focuses characteristics (attitudes, life stages, scenarios, etc.) that are most relevant to the person's experiences in managing financial and health-related concerns.

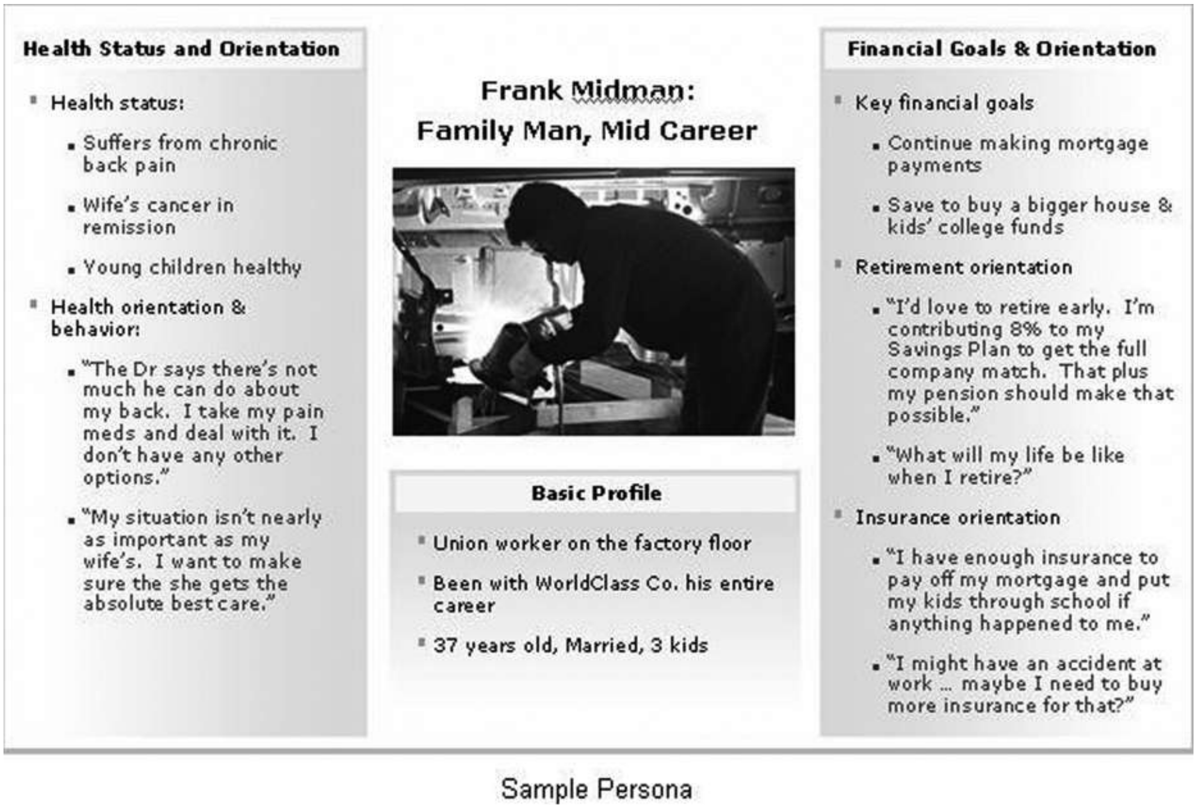
The value of personas can be enhanced by making them visible and dynamically present for design and development teams (e.g., posters displayed in project rooms, multimedia representations that are reviewed with development teams, role-playing scenarios and walkthroughs based on profiles, etc.). Rich and dynamic representations of essential characteristics of individuals can serve as a common frame of reference for communication and a tangible reminder to development teams regarding the people for whom they are designing the system. Moreover, personas can be used systematically in a range of ways to help teams make design decisions. For example, Pruitt and Grudin (2003) described specific techniques they have used to systematically apply personas to aid in feature prioritization decisions.

45.17.2.4 Scenarios

Scenarios are another way ethnographic research findings can be portrayed (Carroll 2000; Nardi 1992; Sonderegger et al. 2000; Chapter 48 of this book). Scenarios illustrate experiences and actions as they unfold in specific contexts or situations (Figure 45.5) and can be documented in various forms ranging from narratives to annotated visual flow diagrams. They may highlight interactions (with computer systems, people, business entities, etc.), decisions processes, activity sequences, influencing factors, and so forth. They also may illustrate the different ways in which varied groups or types of people experience and navigate through similar situations. Analysis of scenarios can foster the identification of areas of difficulty ("pain points") and experiential gaps (or opportunities), that may be addressed or enhanced through various design solutions. When integrated with personas, they can illustrate how different target audiences navigate through the same situation, which in turn can suggest ways in which solutions can and should be adapted for varying target audiences.

45.17.2.5 Service Blueprints

As service design has become a more important arena for the application of ethnographic approaches, designers and researchers have looked for ways of representing services.



Health Status and Orientation

- Health status:
  - Suffers from chronic back pain
  - Wife's cancer in remission
  - Young children healthy
- Health orientation & behavior:
  - "The Dr says there's not much he can do about my back. I take my pain meds and deal with it. I don't have any other options."
  - "My situation isn't nearly as important as my wife's. I want to make sure the she gets the absolute best care."

Frank Midman: Family Man, Mid Career

Basic Profile

- Union worker on the factory floor
- Been with WorldClass Co. his entire career
- 37 years old, Married, 3 kids

Financial Goals & Orientation

- Key financial goals
  - Continue making mortgage payments
  - Save to buy a bigger house & kids' college funds
- Retirement orientation
  - "I'd love to retire early. I'm contributing 8% to my Savings Plan to get the full company match. That plus my pension should make that possible."
  - "What will my life be like when I retire?"
- Insurance orientation
  - "I have enough insurance to pay off my mortgage and put my kids through school if anything happened to me."
  - "I might have an accident at work … maybe I need to buy more insurance for that?"

Sample Persona

FIGURE 45.4 Financial and health benefit program design persona.

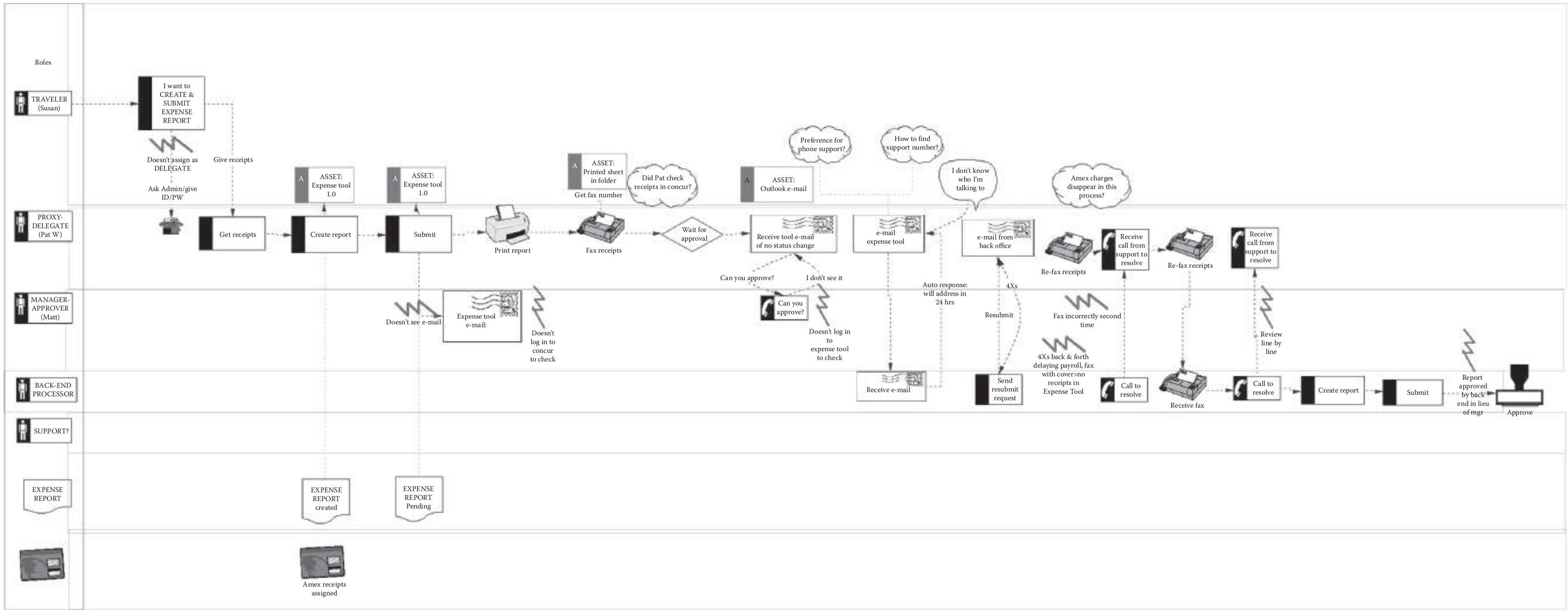

Roles
TRAVELER (Susan)
PROXY-DELEGATE (Pat W)
MANAGER-APPROVER (Matt)
BACK-END PROCESSOR
SUPPORT?
EXPENSE REPORT
I want to CREATE & SUBMIT EXPENSE REPORT
Doesn't assign as DELEGATE
Ask Admin/give ID/PW
Give receipts
Get receipts
A ASSET: Expense tool 1.0
Create report
A ASSET: Expense tool 1.0
Submit
Doesn't see e-mail
Expense tool e-mail:
Doesn't log in to concur to check
Print report
A ASSET: Printed sheet in folder
Get fax number
Fax receipts
Did Pat check receipts in concur?
Wait for approval
A ASSET: Outlook e-mail
Preference for phone support?
Receive tool e-mail of no status change
Can you approve?
I don't see it
Can you approve?
Doesn't log in to expense tool to check
How to find support number?
e-mail expense tool
Receive e-mail
Auto response: will address in 24 hrs
I don't know who I'm talking to
e-mail from back office
4Xs
Resubmit
Send resubmit request
Amex charges disappear in this process?
Re-fax receipts
Fax incorrectly second time
4Xs back & forth delaying payroll, fax with cover>no receipts in Expense Tool
Receive call from support to resolve
Call to resolve
Re-fax receipts
Receive fax
Call to resolve
Review line by line
Receive call from support to resolve
Create report
Submit
Report approved by back end in lieu of mgr
Approve
EXPENSE REPORT created
EXPENSE REPORT Pending
Amex receipts assigned

FIGURE 45.5 Scenario flow model.

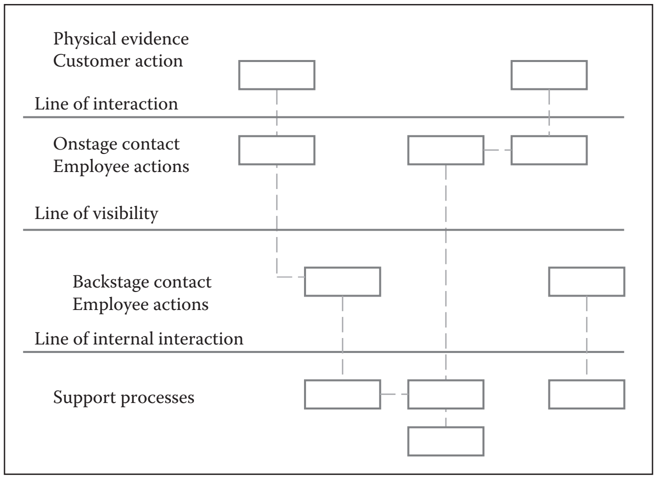


FIGURE 45.6 Service blueprint elements.

Service Blueprinting is a service design tool and a way of representing people, interactions, organizations, and artifacts that make up a service. The approach first introduced to enable innovation in services (Shostack 1984, 1987, and 1993) has evolved and become a standard way of describing services, introducing change in how the service is delivered, or designing radically new services. Service blueprinting was initially introduced as a process control technique (Bitner, Ostrom, and Morgan 2008) for services that provided more precision than verbal descriptions and could therefore identify possible failure points. Service blueprinting is now focused on improving the service experience, noting just those places where the service recipient comes into contact with the provider or provider organization. Central to the service blueprinting representation is a distinction between front stage and back stage or those aspects of the service that involve or require interaction with the customer and those that occur behind the scenes, in the provider organization. Importantly service blueprinting highlights the customer's role in providing the service and as such focuses attention on the experiences of "users" of the service. Service blueprinting shares similarities to both process models and scenarios in that the representation focuses on the often linear unfolding of a service (e.g. from the time you walk into a bank, approach the teller, complete your transaction, and depart). Service blueprinting allows service designers to focus on where the customer interfaces with the service provider and to connect those interactions with the activities taking place outside their view, but nonetheless critical to providing the service (see Figure 45.6).

45.17.2.6 Mental Models

The concept of a "mental model" has a long history in cognitive science and has been utilized in a variety of ways in HCI and interaction design (e.g., Gentner and Stevens 1983; Johnson-Laird 1983, 1996). For example, Norman (1983) used the term mental model to refer to the "internal conceptualizations that people form of the things with which they interact." More broadly, mental models can be defined as representations of how people make sense of and think about an experience or a product or service. Although ethnography does not enable one to directly "observe" how people create meaning, it does aim to develop views onto how the people studied understand and make sense of their world. In the current context, interpretation and representation of ethnographic findings in the form of mental models can inform the design of products and services by highlighting the key conceptual facets of experiences, the categories that people apply, the questions they ask, the principles they apply, the ways they think and flow through problems and information. The insights reflected in such models can help shape numerous aspects of a product or service including the organization and architecture, the nomenclature and labels, and the interaction design.

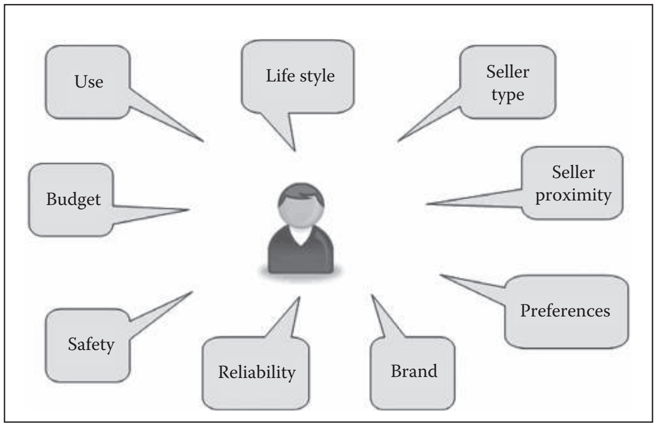


FIGURE 45.7 Simple metal model of car buying criteria.

For example, a simple mental model representing the key questions and criteria that people apply when looking for a car (see Figure 45.7) can inform the entry points and exploration criteria (e.g., faceted navigation categories and criteria) presented to users of an automotive classified website. A mental model representing the primary and secondary questions that design engineers think about when selecting and designing parts or assemblies for new vehicles (see Figure 45.8) can similarly inform the design of a discovery application aimed at helping find the best parts for their products. Engineers are aided in their exploration and decision making through the selection and presentation of important data sources organized by meaningful categories such as form, fit, function, environmental compliance, cost, reliability, and lifespan. These provide search and discovery entry points and refinement and exploration criteria.

45.17.2.7 Mock-Ups and Prototypes

Representational artifacts, be they paper prototypes, mock-ups, or working prototypes, can play an important mediating role in connecting use requirements and design possibilities. When informed by studies of practice, these design representations respecify practices and activities in ways that are recognizable to practitioners. The prototypes go beyond simple

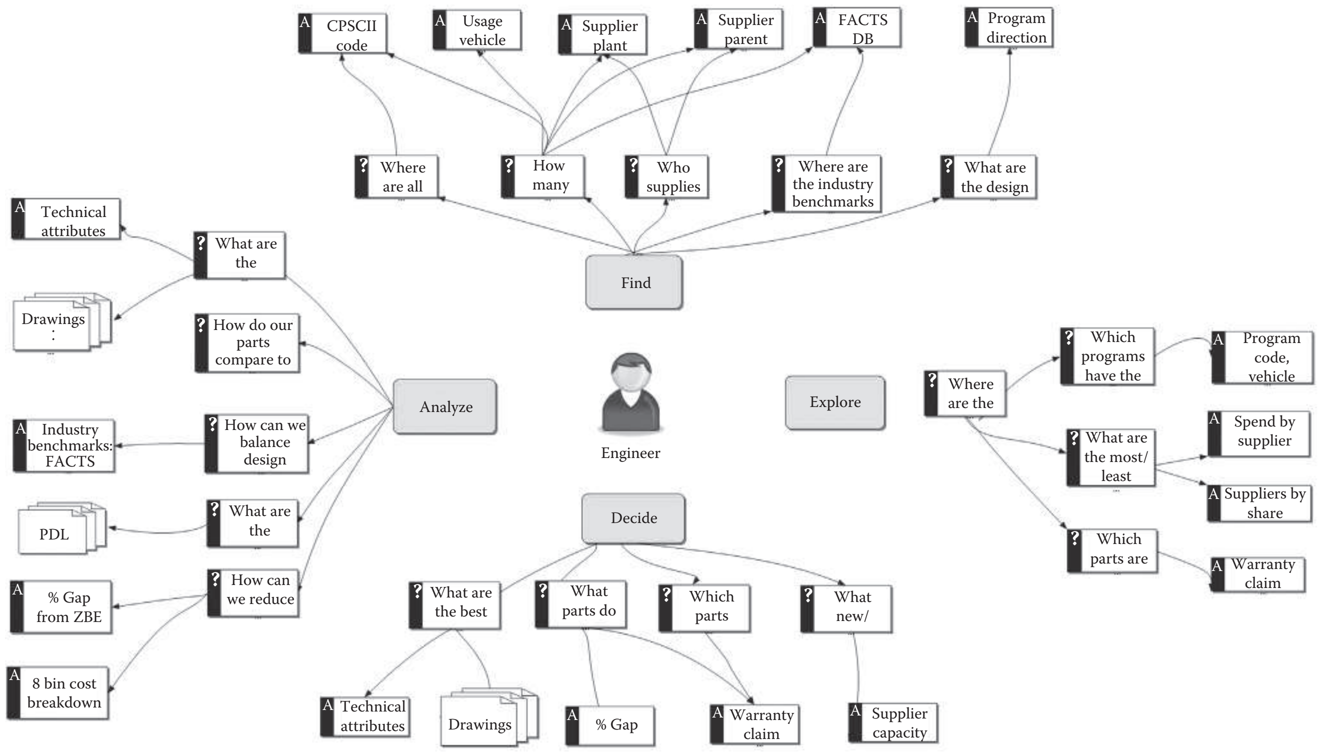


FIGURE 45.8 Design engineers' mental model for selecting and designing new vehicles.

demonstrations of functionality to incorporate materials from participants' sites, embody envisioned new technological possibilities, convey design ideas in relation to existing practices, and reveal requirements for new practices. Prototyping practices as such recover and invent use requirements and technological possibilities that make sense each in relation to the other (Suchman, Blomberg, and Trigg 2002). In addition these representational artifacts facilitate the communication of what has been learned about technologies-in-use to the larger research and technology-development communities.

In an ethnographic study of engineering practice at a state Department of Highways, design prototypes critically deepened the researchers understanding of the requirements of the work of document filing and retrieval (the focus of the study). At each step, from early design discussions with practitioners, to the creation of paper "mock-ups" of possible interfaces to the online project files, and finally to installing a running system at the worksite, the researchers became more aware of the work's exigencies. For example, in recognition of some of the difficulties that engineers experienced with their filing system various alternative document-coding strategies that augmented the existing filing system were designed. Through successive rounds, in which engineers were asked to code documents using mocked-up coding forms (both paper-based and online) the researchers' understanding of the requirements of the work deepened. Eventually, the search and browsing interfaces evolved to be more finely tuned to the requirements of the engineers' work (e.g., Trigg, Blomberg, and Suchman 1999).

45.17.3 Caveat Regarding Representations and Models

Although representations and models are valuable tools for connecting ethnographic understanding and design, they also can have negative effects. Although grounded in observations and other forms of ethnographic inquiry, models are always a selective interpretation and construction of experience. Thus, while representations and models can focus attention on and illuminate important aspects of experience, they can also become reified stereotypes and constraints that inhibit design possibilities. Ongoing inquiry, a critical perspective, and a willingness to evolve the representations in the face of new learning are essential to maintain the viability and value of models for design.

45.18 RELATION TO OTHER QUALITATIVE APPROACHES AND PERSPECTIVES

The ethnographic approach has strong connections to and affinities with other approaches that have contributed to the development of the field of HCI, namely distributed cognition, activity theory, ethnomethodology, and participatory design. In addition, the connections between ethnography and usability testing have grown stronger in recent years with some innovative approaches to combining user testing with observations and interviewing in more naturalistic settings. There is not space here to go into depth on any of these approaches. Our aim is simply to highlight relations between these approaches and ethnography, and provide a way to distinguish between them.

Distributed cognition (sometimes referred to as social or situated cognition) was first introduced to the HCI community by Lave (1988) and Hutchins (1995). Distributed cognition located cognition in social and material processes. When it was introduced, it challenged the dominant paradigm within HCI, that cognition primarily involved the psychological and mental processes of individuals. The connection between distributed cognition and ethnography is not only in the insistence that our understanding of human activity be located outside individual mental processes, in human interaction, but also in the conviction that to gain an understanding of human activity, ethnographic, field-based methodologies are required.

Activity theory also shares with ethnography a commitment to field-based research methodologies. In addition, there is the shared view that behavior (activity) should be a primary focus of investigation and theorizing, and a recognition that objects (artifacts) are key components in descriptive and explanatory accounts of human experience (e.g., Engeström 2000; Nardi 1996).

Ethnomethodology is often used interchangeably with ethnography in HCI literature. This is not only because the terms are etymologically similar, but also because many of the social scientists contributing to the field of HCI have adopted an ethnomethodological approach (e.g., Bentley et al. 1992; Button and Harper 1996; Crabtree 2000; Hughes, Randall, and Shapiro 1993; Hughes et al. 1994; Hughes, Rodden, and Anderson 1995) with its focus on locally and interactionally produced accountable phenomena. Ethnomethodology's particular set of commitments (e.g., Heritage 1984) are not shared however by everyone working within the ethnographic paradigm.

Participatory design does not have its roots in qualitative social science research, but instead developed as a political and social movement, and as a design approach committed to directly involving end users in the design of new technologies (See Chapter 49; also Schuler and Namioka [1993]; Kensing and Blomberg [1998]). Within the HCI context, participatory design has shed some of its political and social-action underpinnings, and often is viewed primarily as a set of methods and techniques for involving users in design. Its connection to ethnography is in the commitment to involve study participants in the research, and in the value placed on participants' knowledge of their own practices. Also in recent years, those working in the field of participatory design have incorporated ethnographic techniques (e.g., Crabtree 1998; Kensing, Simonsen, and Bødker 1999) as a way of jointly constructing with participants knowledge of local practices.

Traditional usability testing, with an emphasis on controlled studies and directed scenario-task based testing of design artifacts is often regarded as antithetical to an ethnographic approach. However, over the years, many have begun to "reframe" usability testing (Buur and Bødker 2000) and integrate methods and approaches derived from ethnography and participatory design into usability testing, blurring the boundaries. For example, Kantner, Sova, and Rosenbaum (2003) describe "field usability testing," conducting testing in context with users in their own environments, working on their own goals with their own artifacts and "task objects" as a means of learning about people's everyday activities and needs, as well as gaining insight into how people might interact with and use complex products in the contexts of their own lives. The continued evolution of pervasive and mobile computing provides additional incentive to explore how people interact with products and services in context. Field testing and virtual observations and interactions with people as they use technologies in the context of their ongoing mobile routines and activities (e.g., Gallant 2006), may actually become the norm as interacting with stationary artefacts becomes a smaller part of the HCI ecosystem and as recording and communication devices become increasingly ubiquitous (e.g., mobile phones equipped with video cameras and GPS tracking capabilities).

The key lesson for both "ethnographers" and "usability testers" is that combining observing and inviting users to interact with design artifacts in the context of their natural environments and everyday activities can be both efficient and valuable. Balancing open ended exploration and observation with personally meaningful, in context scenario-based assessments can yield extremely useful insights about how to make products and services more effective and "usable," how people operate and make sense of human computer interfaces, how people improvise and use new technologies and solutions in unexpected ways, and how potential barriers and facilitators shape user adoption.

45.19 ETHNOGRAPHY IN ACTION

The following two case studies show how an ethnographic approach was applied in the design of a program to change health-related behaviors and to reconfiguring service interactions in IT outsourcing services. The two cases point to the role of ethnographic research in rethinking basic assumptions about what motivates and enables employee choices in one case and the place of IT performance data in building and sustaining client-provider relationships in the other. In addition, specific design recommendations followed from these two studies.

45.19.1 Case Study 1: Designing a Program and Website to Change Health Care Behaviors

A large global company, providing health insurance coverage to over 60,000 of its employees in the United States, developed a multifaceted program to reduce its health care costs and optimize the health and productivity of its workforce. The major goals were to provide reliable health care information and to promote better health care decisions. The program provided a number of online and offline resources for employees (e.g., a 24-hour medical hotline, a research team that would provide gather and summarize treatment outcome research findings for severe medical conditions, online access to a leading edge medical information/content website, etc.). The company initially promoted the program through a series of face-to-face workshops designed to convey the limitations

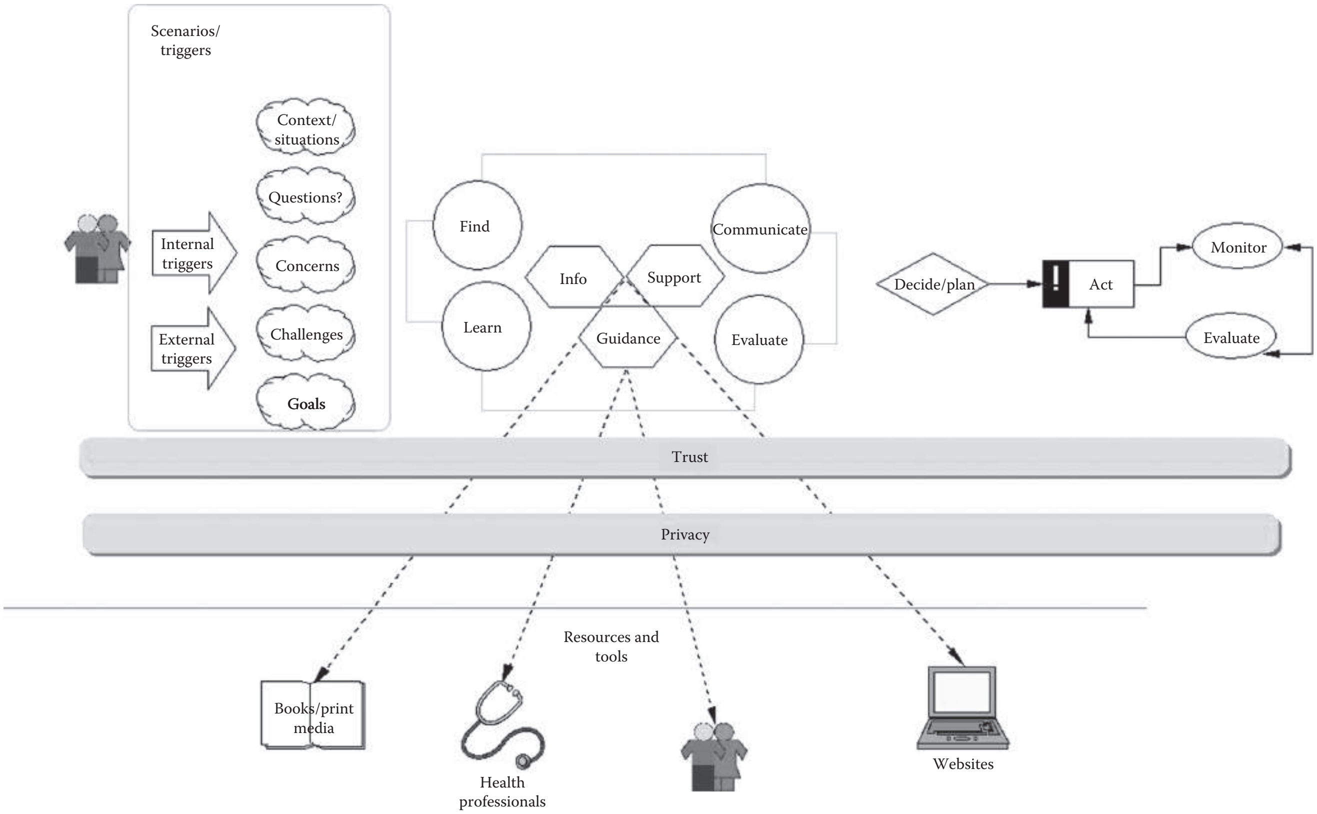


FIGURE 45.9 Healthcare decision-making model.

of standard medical practice, encourage a consumer-oriented approach to health care, and make people aware of resources provided by the company.

After the initial launch, the team became concerned that the health program resources, including the website were being underutilized by employees, limiting the potential impact and value for both employees and the company. In this context, the team initiated a study to evaluate the current program and website as well as to establish clear user models and strategic frameworks to guide website/program redesign efforts. To meet the project objectives, the research team conducted ethnographic inquiries combined with scenario-based exercises. In order to extend the participant sample as well as deal with practical constraints (very limited time and resources), the team conducted some of the interviews and assessments remotely, via telephone and web conferencing tools. Ethnographic inquiries focused on understanding the varied ways that people managed their health care (and/or the health care of family members), including their overall orientations to health and wellness, relationships and interactions with health care providers (and other family members), and their health care–decision-making processes. The latter included understanding the online and offline resources and tools that people used and the major health care scenarios they addressed. After exploring and profiling participants' health care experiences, they were asked to work through an actual health care decision scenario, while being invited to engage with the program resources and website.

Based on these inquiries, the research team developed a number of experience models including a set of personas highlighting key variations in health care orientation and behavior that the program/website design team would have to accommodate; a simple typology of health-related scenarios (e.g., managing severe and chronic medical conditions, dealing with common everyday health care issues, and "wellness"/risk reduction); scenario flow models documenting how varied types of people made decisions (Figure 45.9) and used a range of resources to address key health scenarios.

These models along with other resources generated numerous insights about limitations of the current website and program, opportunities for program/website enhancement, and design recommendations. For example, user profiles and scenario models showed how the program was fragmented and did not effectively align with people's key health scenarios, forcing an individual to painfully sift through resource information and descriptions to figure out which resources might be most relevant and useful in a specific scenario. In addition, the program and the website did not adequately address "wellness"/risk reduction scenarios which represented a significant concern for almost all employee segments and presented an important opportunity for the company to promote a proactive and preventative approach to healthcare.

In order to connect the user insights with the program/website design, the team articulated a number of design principles and a specific scenario-based design framework (Figure 45.10). This framework highlighted the value of

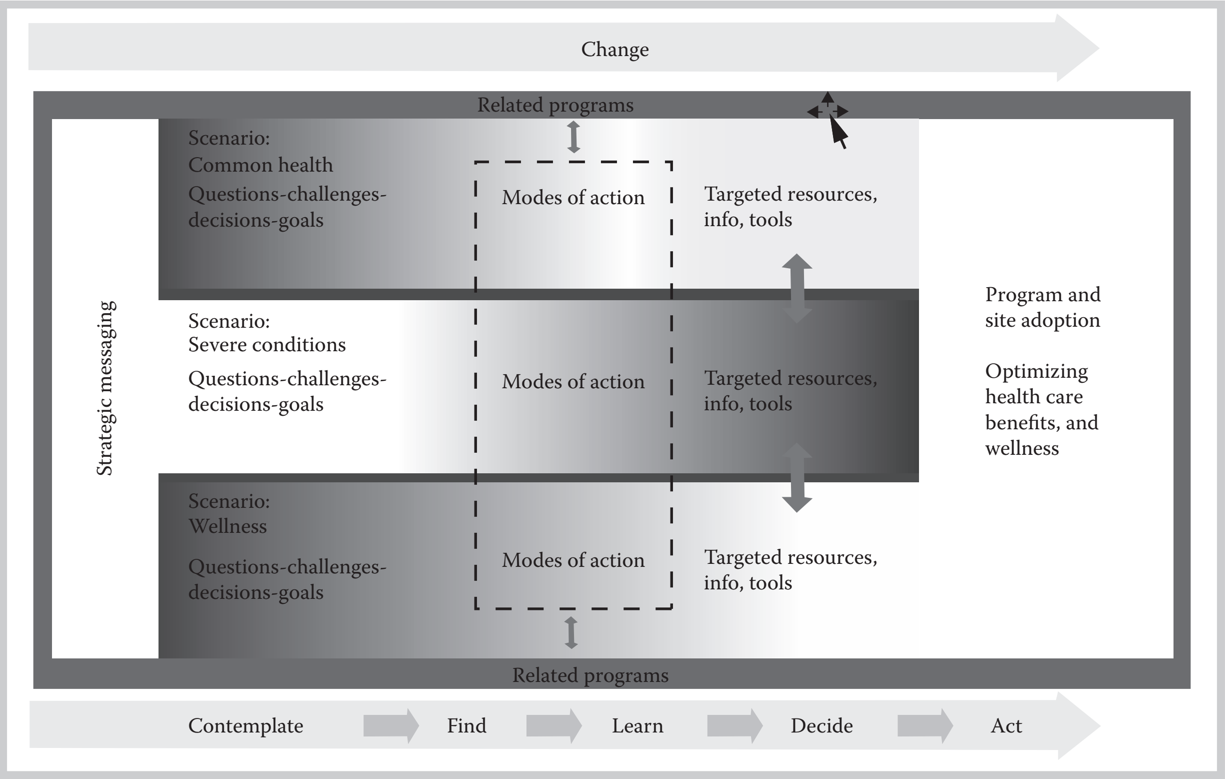

Change
Related programs
Scenario:
Common health
Questions-challenges-decisions-goals
Modes of action
Targeted resources, info, tools
Strategic messaging
Scenario:
Severe conditions
Questions-challenges-decisions-goals
Modes of action
Targeted resources, info, tools
Program and site adoption
Optimizing health care benefits, and wellness
Scenario:
Wellness
Questions-challenges-decisions-goals
Modes of action
Targeted resources, info, tools
Related programs
Contemplate
Find
Learn
Decide
Act

FIGURE 45.10 Sample experience design framework.

organizing the website (and other program elements) based on key healthcare scenarios, aligning and prioritizing resources and inviting specific modes of action that were most important in each scenario, and enabling relevant "cross-scenario" awareness and behavior that would provide value to users and support program objectives (e.g., a person who came to the website to learn about current research findings on the effectiveness of an experimental treatment for diabetes, might also be invited to explore the value of changes in diet or exercise to manage diabetes, etc.). In addition, the framework highlighted the importance of embedding strategic messages regarding health care (e.g., importance of evidence-based medicine, proactively taking charge of one's health and health care, etc.) and implicit invitations to change health care behaviors throughout the site design.

The ethnographic research led the team to rethink a number of major assumptions, which in turn led to redesign of program strategies, resources, and the website. From a program perspective, the research highlighted the fact that the vast majority of employees had already adopted many consumer attitudes and behaviors and were leveraging a number of trusted health resources (in contrast to initial assumptions of limited "consumerism"). This led the team to reconsider the positioning of specific program resources, shift strategic messaging, and generate novel program strategies including behavioral "rewards" programs that supported proactive and preventative behaviors. The initial research inquiry also led the team to implement a continuous assessment program to continue to monitor program impact and changes in employee experiences and behaviors.

45.19.2 Case Study 2: Service Provider-Client Interactions: Enabling Sustainable Relationships in IT Outsourcing Services

The design of a web-based portal to facilitate interactions between service providers and their clients (see Figure 45.11) was the focus of an ethnographic study examining the communication and knowledge sharing practices of executive level employees in both the service provider and the client organizations (Blomberg 2008a). This research was directed at service innovation to establish and maintain sustainable client-provider relationships in IT outsourcing services. In particular the portal development effort was initiated to

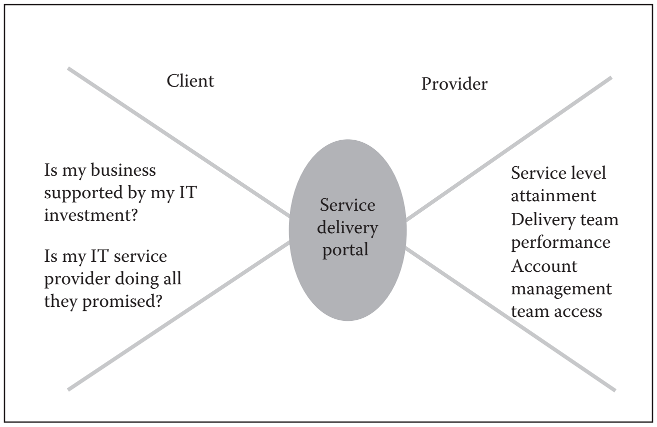

Client
Provider
Is my business supported by my IT investment?
Is my IT service provider doing all they promised?
Service delivery portal
Service level attainment
Delivery team performance
Account management team access

FIGURE 45.11 Portal facilitated client-provider interactions.

increase client access to service performance metrics and, in so doing, allow greater information transparency (Blomberg 2008b). A primary way in which clients and providers track IT performance is through Service Level Agreements (SLAs), which detail contractual relationships between service provider and client, and describe the metrics that will be used to regulate and monitor the performance of the delivery teams (Long-Tae, Jong-Wook, and Woon-Ki Hong 2001; Marilly et al. 2002). The SLA specifies the level of service the customer can expect from the provider. Service Level Attainment metrics are tracked each month, and failure to perform as expected can result in penalties that the provider must pay. Typically the service provider is obliged to provide monthly reports that describe actual performance metrics in relation to the attainment levels specified in SLA. A key element of the initial portal design was the provision for presenting to the client "real-time" IT performance metrics as detailed in the SLA. This design direction led to an ethnographic study on how IT service performance metrics were currently made available to executive level clients.

Interviews and observations of IT executives from the provider and client organizations highlighted how performance information was communicated via either face-to-face meetings or teleconferences where executives reviewed the performance numbers and arrived at "negotiated" understandings of what accounted for the numbers and what should be done to address any shortcomings. Through these interactions service providers and clients arrived at the "meanings" of the performance metrics which enabled them to develop both immediate and longer term plans to mitigate problems and, as appropriate, expand the scope of the engagement.

It became clear that providing accurate, real-time performance information alone would do little to enable effective communication and might instead undermine trust between executives from the client and provider organizations. For example, knowing that response times for help desk calls were behind targets was not as important as understanding the causes of the slower response times and having confidence that steps were being taken to address the situation. Logging onto the portal to access performance information without the opportunity to understand the meanings behind these data could create confusion and unnecessary concern on the part of the client.

This research finding led to the recommendation that real-time interactions between clients and providers be explicitly supported, including during face-to-face meetings. The design specification had not included provisions for downloading performance data to a spreadsheet or other applications so that it could be easily shared and referenced in meetings between the executives. It was in these interactions that the meanings of performance metrics were negotiated and changes to address performance issues were agreed.

The study also showed how IT performance measures and reporting formats evolved over time in response to adjustments in the service contract or in response to requests from the client to visualize performance data in different formats (e.g. bar charts instead of, or in addition to, tables). These changes facilitated discussions between providers and clients when specific IT decisions were under deliberation. The providers were motivated to comply with these client requests, even if they were not specified in the contract, to strengthen their relationship with the client and ultimately the long term health of the account. This led to the recommendation that changing the way performance data were presented should be within the control of the service delivery teams. The design specification had not included this capability, instead requiring that the code be rewritten by those who developed the original portal.

In these ways our design recommendations centered on enhancements to the portal technology that would better support interactions that facilitated the negotiation of the meaning of performance data, enabled changes in the reporting needs of the client and provider, and more seamlessly integrated the portal reporting format and interactions taking place face-to-face or via teleconference. The study showed that making performance metrics available alone would not achieve the objectives of portal no matter how accurate the data were and how accessible they were in real time.

45.20 CONCLUSION

Ethnographic studies have become an important tool for designers and development teams designing new information and communication technologies and new IT-enabled services. Today in academic, institutional, and corporate settings there is the realization that understanding the everyday realities of people living and working in a wide range of environments and engaged in a myriad of activities is essential for creating technologies and services that provide engaging and productive experiences for their users.* Emerging from recent research and practical experience is the recognition that representational tools (models, personas, scenarios, mock-ups and prototypes, service blueprints, etc.) and design-and-development practices (collaborative data analysis, video review sessions, etc.) are necessary for connecting ethnographic studies and technology design. Insights from ethnographic studies do not map directly onto design specifications or straightforwardly generate "user" requirements. Instead ethnographic studies must be connected and integrated with design agendas and practices. Those wishing to leverage the potential of ethnographic studies should not only understand what motivates the approach and is at its foundation (e.g., natural settings, holistic, descriptive, members' point of view), but also should recognize the importance of creating the conditions in which design can take advantage of ethnographic insights.

* For a discussion of the relation between ethnography and design, see also Anderson (1994), Grudin and Grintner (1995), Rogers and Bellotti (1997), and Shapiro (1994).

REFERENCES

Agar, M. 1996. *The Professional Stranger*. 2nd ed. San Diego, CA: Academic Press.

Anderson, K., and T. Lovejoy. 2005. *Proceedings of EPIC 2005*. Berkeley: University of California Press.

Anderson, R. J. 1994. Representations and requirements: The value of ethnography in system design. *Hum Comput Interact* 9:151–82.

Appadurai, A., ed. 1988. *The Social Life of Things: Commodities in Cultural Perspective*. Cambridge: Cambridge University Press.

Babbie, E. 1990. *Survey Research Methods*. 2nd ed. Belmont, CA: Wadsworth Publishing Company.

Bell, G. 2004. Insights into Asia: 19 Cities, 7 Countries, 2 Years—What people Really Want from Technology. Technology@ Intel Magazine. Intel Corp.

Bentley, R., J. A. Hughes, D. Randall, T. Rodden, P. Sawyer, D. Shapiro, and I. Sommerville. 1992. Ethnographically-informed system design for air traffic control. In *Proceedings of Computer Supported Cooperative Work*, 123–9. New York: ACM Press.

Bernard, H. R. 1995. *Research Methods in Anthropology: Qualitative and Quantitative Approaches*. 2nd ed. London: Altamira Press.

Berry, M., and M. Hamilton. 2006. Mobile computing, visual diaries, learning and communication: Changes to the communicative ecology of design students through mobile computing. In *Proceedings of the 8th Australian Conference on Computing Education*. Darlinghurst, Australia: Australian Computer Society.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: Defining Customer-Centered Systems*. San Francisco, CA: Morgan Kaufmann Publishers.

Bitner, M. J., A. L. Ostrom, and F. N. Morgan. 2008. Service blueprinting: A practical technique for service innovation. *Calif Manage Rev* 50:3.

Blomberg, J. 1987. Social interaction and office communication: Effects on user's evaluation of new technologies. In *Technology and the Transformation of White Collar Work,* ed. R. Kraut, 195–210. Hillsdale, NJ: Lawrence Erlbaum Associates.

Blomberg, J. 1988. The variable impact of computer technologies on the organization of work activities. In *Computer-Supported Cooperative Work: A Book of Readings,* ed. I. Greif, 771–82. San Mateo, CA: Morgan Kaufmann.

Blomberg, J. 1995. Ethnography: Aligning field studies of work and system design. In *Perspectives on HCI: Diverse Approaches*, ed. A. F. Monk and N. Gilbert, 175–97. London: Academic Press LTD.

Blomberg, J. 2008a. On participation and service innovation. In *(Re-) Searching a Digital Bauhaus*, ed. T. Binder, J. Löwgren, and L. Malmborg, 121–44. London: Springer.

Blomberg, J. 2008b. Negotiating meaning of shared information in service system encounters. *Eur Manag J* 23:213–22.

Blomberg, J., J. Giacomi, A. Mosher, and P. Swenton-Wall. 1991. Ethnographic field methods and their relation to design. In *Participatory Design: Perspectives on Systems Design,* ed. D. Schuler and A. Namioka, 123–55. Hillsdale, NJ: Lawrence Erlbaum Associates.

Blomberg, J., L. Suchman, and R. Trigg. 1996. Reflections on a work-oriented design project. *Hum Comput Interact* 11:237–65.

Blomberg, J., and R. Trigg. 2000. Co-constructing the relevance of work practice for CSCW Design: A case study of translation and mediation. Occasional Papers from the Work Practice Laboratory. *Blekinge Inst Technol* 1:1–23.

Bowers, J., G. Button, and W. Sharrock. 1995. Workflow from within and without: Technology and cooperative work on the print industry shopfloor. In *Proceedings of the Fourth Conference on European Conference on Computer-Supported.* Stockholm, Sweden. 51–66. Dordrecth: Kluwer Academic Publishers.

Briggs, C. 1983. *Learning How to Ask: A Sociolinguistic Appraisal of the Role of the Interview in Social Science Research*. Cambridge, U.K.: Cambridge University Press.

Brun-Cotton, F., and P. Wall. 1995. Using video to re-present the user. *Commun ACM* 38:61–71.

Button, G., and R. Harper. 1996. The relevance of 'work-practice' for design. *Comput Support Coop Work* 5:263–80.

Buur, J., and S. Bødker. 2000. From usability lab to "design collaboratorium": Reframing usability practice. In *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*. 297–307. Brooklyn, NY: ACM.

Carroll, J. M. 2000. *Making Use: Scenario-Based Design of Human-Computer Interactions*. Cambridge, MA: MIT Press.

Carter, S., and J. Mankoff. 2005. When participants do the capturing: The role of media in diary studies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 899–908. Portland, Oregon. ACM.

Cefkin, M., ed. 2009. *Ethnographers at Work: New Social Science Research In and of Industry*. New York: Berghahn Books.

Clifford, J. 1988. *The Predicament of Culture: Twentieth-Century Ethnography, Literature, and Art*. Cambridge, MA: Harvard University Press.

Clifford, J., and G. Marcus, eds. 1986. *Writing Culture: The Poetics and Politics of Ethnography*. Berkeley: University of California Press.

Cohen, K. 2005. Who we talk about when we talk about users. In *Proceedings of EPIC*, 9–30. Arlington, VA: American Anthropological Association.

Coleman, S., and P. von Hellermann, eds. 2011. *Multi-Sited Ethnography: Problems and Possibilities in Translocation of Research Methods*. New York: Routledge.

Comaroff, J., and J. Comaroff. 1992. *Studies in the Ethnographic Imagination*. Boulder, CO: Westview Press.

Corral-Verduga, V. 1997. Dual 'realities' of conservation behavior: Self reports vs. observations of re-use and recycling behavior. *J Environ Psychol* 17:135–45.

Crabtree, A. 1998. Ethnography in participatory design. In *Proceedings of the Participatory Design Conference*, 93–105. Seattle, WA: Palo Alto, CA: CPSR.

Crabtree, A. 2000. Ethnomethodologically informed ethnography and information system design. *J Am Soc Inf Sci* 51:666–82.

D'Andrade, R. G. 1995. *The Development of Cognitive Anthropology*. Cambridge, U.K.: Cambridge University Press.

Dittrich, Y., A. Ekelin, P. Elovaara, S. Eriksén, and C. Hansson. 2003. Making e-Government happen Everyday co-development of services, citizenship and technology. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 5*.

Dourish, P. 2001. Process descriptions as organizational accounting devices: The dual use of workflow technologies. In *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, 52–60. Boulder, CO.

Engeström, Y. 2000. From individual action to collective activity and back: Developmental work research as an interventionist methodology. In *Workplace Studies: Recovering Work Practice and Informing System Design,* ed. P. Luff, J. Hindmarsh, and C. Heath, 150–66. Cambridge, U.K.: Cambridge University Press.

Falzon, M. 2009. *Multi-Sited Ethnography: Theory, Praxis and Locality in Contemporary Research*. Aldershot: Ashgate Pub.

Fluehr-Lobban, C., ed. 1991. *Ethics and the Profession of Anthropology: Dialogue for a New Era*. Philadelphia: University of Pennsylvania Press.

Foucault, B. E., R. S. Russell, and G. Bell. 2004. Techniques for researching and designing global products in an unstable world: A case study. In *CHI'04 Extended Abstracts on Human Factors in Computing Systems*. New York: ACM.

Gallant, L. M. 2006. An ethnography of communication approach to mobile product testing. *Pers Ubiquitous Comput* 10(5):325–32.

Gentner, D., and A. Stevens. 1983. *Mental Models*. New York: Lawrence Ehrlbaum Associates.

Gillham, R. 2005. Diary studies as a tool for efficient cross-cultural design. In *Proceedings of International Workshop on Internationalisation of Products and Services (IWIPS)*, 57–65. Amsterdam, The Netherlands.

Green, E. C. 2001. Can qualitative research produce reliable quantitative findings? *Field Methods* 13:1–19.

Green, S., P. Harvey, and H. Knox. 2005. Scales of place and networks: An ethnography of the imperative to connect through information and communication technologies. *Hum Organiz* 46:805–26.

Grief, I., ed. 1988. *Computer-Supported Cooperative Work: A Book of Readings*. San Mateo, CA: Morgan Kaufmann.

Grudin, J., and R. E. Grintner. 1995. Ethnography and design. *Comput Support Coop Work* 3:55–9.

Gubrium, J. F., and J. A. Holstein, eds. 2002. *Handbook of Interview Research: Context and Method*. Thousand Oaks, CA: Sage Publication.

Guest, G. 2000. Using Guttman scaling to rank wealth: Integrating quantitative and qualitative data. *Field Methods* 12:346–57.

Hannerz, U. 2003. Being there…and there…and there! *Ethnography* 4:201–16.

Harding, S. 1986. *The Science Question in Feminism*. Ithaca, NY: Cornell University Press.

Heritage, J. 1984. *Garfinkel and Ethnomethodology*. Cambridge, MA: Polity Press.

Horst, H., and D. Miller. 2005. From kinship to link-up: Cell phones and social networking in Jamaica. *Hum Organiz* 46:755–78.

Hughes, J. A., D. Randall, and D. Shapiro. 1993. From ethnographic record to system design: Some experiences from the field. *Comput Support Coop Work* 1:123–47.

Hughes, J., V. King, T. Rodden, and H. Anderson. 1994. Moving out of the control room: Ethnography in systems design. In *Proc. CSCW'94*, 429–438. Chapel Hill, NC: ACM Press.

Hughes, J. A., T. Rodden, and H. Anderson. 1995. The role of ethnography in interactive system design. *ACM Interact* 2:56–65.

Hutchins, E. 1995. *Cognition in the Wild*. Cambridge, MA: MIT Press.

Johnson, J. C. 1990. Selecting ethnographic informants. Newbury Park, CA: Sage.

Johnson, J. C., and D. C. Griffith. 1998. Visual data: Collection, analysis, and representation. In *Using Methods in the Field: A Practical Introduction and Casebook,* ed. V. DeMunck and E. Sobo, 211–28. Walnut Creek, CA: Altamira.

Johnson, J. C., M. Ironsmith, A. L. Whitcher, G. M. Poteat, and C. Snow. 1997. The development of social networks in preschool children. *Early Educ Dev* 8:389–406.

Johnson-Laird, P. N. 1983, 1996. *Mental Models*. Cambridge: Harvard University Press.

Jones, M., and F. Samalionis. 2008. From small ideas to radical service innovation. *Des Manage Rev* 19:20–7.

Jordan, B., and L. Suchman. 1990. Interactional troubles in face-to-face survey interviews. *J Am Stat Assoc* 85(409):232–53.

Kantner, L. 2001. Assessing website usability from server log files. In *Design by People, for People: Essays on Usability*, ed. R. Branaghan, 245–62. Chicago, IL: Usability Professionals Association.

Kantner, L., D. H. Sova, and S. Rosenbaum. 2003. Alternative methods for field usability research. In *Proceedings SIGDOC Annual International Conference on Documentation*, 68–72. San Francisco, CA.

Karasti, H. 2001. Bridging work practice and system design—integrating systemic analysis, appreciative intervention, and practitioner participation. *Comput Support Coop Work Int J* 10:167–98.

Kensing, F., and J. Blomberg. 1998. Participatory design: Issues and concerns. *Comput Support Coop Work* 7:163–5.

Kensing, F., J. Simonsen, and K. Bødker. 1999. MUST—a method for participatory design. *Hum Comput Interact* 13:167–98.

Kieliszewski, C. A., J. H. Bailey, and J. Blomberg. 2010. A service practice approach: People, activities and information in highly collaborative knowledge-based service systems. In *Handbook of Service Science*, ed. P. P. Maglio, C. A. Kieliszewski, and J. C. Spohrer. U.S.: Springer.

Kimbell, L. 2009. The turn to service design. In *Design and Creativity Policy, Management and Practice*, ed. G. Julier and L. Moor, 157–73. Oxford: Berg.

Kimbell, L., and V. P. Seidel. 2008. Designing for Services—Multidisciplinary Perspectives: Proceedings from the Exploratory Project on Designing for Services in Science and Technology-based Enterprises, Saïd Business School. http://www.scribd.com/doc/72888892/Designing-for-Services-Multidisciplinary-Perspectives. Accessed November 30, 2011.

Koskinen, I., K. Battarbee, and T. Mattelmäki. 2005. *Emphathic Design: User Experience in Product Design*. Helsinki: IT Press.

Latour, B. 1987. *Science in Action: How to Follow Scientists and Engineers Through Society*. Cambridge, MA: Harvard University Press.

Latour, B., and S. Woolgar. 1986. *Laboratory Life: The Construction of Scientific Facts*. Princeton, NJ: Princeton University Press.

Lave, J. 1988. *Cognition and Practice*. Cambridge, U.K.: Cambridge University Press.

Leonard, D., and J. F. Rayport. 1997. Sparking innovation through empathic design. *Harv Bus Rev* 75:102–13.

Long-Tae, P., B. Jong-Wook, and J. Woon-Ki Hong. 2001. Management of service level agreements for multimedia Internet service using a utility model. *Commun Mag IEEE* 39:100–6.

Mager, B., and O. King. 2009. Methods and processes of service design. *Touchpoint* 1:20–8.

Malone, T. 1983. How do people organize their desks? Implications for the design of office information systems. *ACM Trans Inf Syst* 1:99–112.

Marcus, G. E. 1995. Ethnography in/of the world system: The emergence of multi-sited ethnography. *Annu Rev0020Anthropol* 24:95–117.

Marcus, G., and M. Fischer. 1986. *Anthropology as Cultural Critique: An Experimental Moment in the Human Sciences*. Chicago: University of Chicago Press.

Marilly, E., O. Martinot, S. Betge-Brezetz, and G. Delegue. 2002. Requirements for service level agreement management. In *IEEE Workshop on IP Operations and Management*, 57–62.

Mason, B., and B. Dicks. 1999. "The digital ethnographer," Cybersociology 6, http://www.cybersociology.com/files/6_1_virtualethnographer.html. Accessed November 30, 2011.

Masten, D., and T. Plowman. 2003. Digital ethnography: The next wave in understanding the consumer experience. *Des Manage J* 14:75–84.

Moore, R. J. 2004. Managing troubles in answering survey questions: Respondents' uses of projective reporting. *Soc Psychol Q* 67:50–69.

Murthy, D. 2008. Digital ethnography: An examination of the use of new technologies for social research. *Sociology* 42:837–55.

Nader, L. 1974. Up the anthropologist—perspectives gained from studying up. In *Reinventing Anthropology,* ed. D. Hymes, 284–311. New York: Vintage.

Nardi, B. 1992. The use of scenarios in design. *SIGCHI Bull* 24:13–4.

Nardi, B. 1996. *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, MA: MIT Press.

Nardi, B., and J. Miller. 1990. An ethnographic study of distributed problem solving in spreadsheet development. In *Proceedings of Computer Supported Cooperative Work*, 197–208. New York: ACM Press.

Nardi, B. A., D. J. Schiano, and M. Gumbrecht. 2004. Blogging as social activity, or, would you let 900 million people read your diary? In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*. New York: ACM.

Newman, S. E. 1998. Here, there, and nowhere at All: Distribution, negotiation, and virtuality in postmodern engineering and ethnography. *Knowl Soc* 11:235–67.

Nielsen, J., and T. K. Landauer. 1993. A mathematical model of the finding of usability problems. In *Proceedings of ACM INTERCHI'93 Conference*, 206–13. Amsterdam, The Netherlands: ACM Press.

Norman, D. 1983. Some observations on mental models. In *Mental Models*, ed. D. Gentner and A. Stevens. Hillsdale, NJ: Lawrence Ehrlbaum Associates.

Palen, L., and M. Salzman. 2002. Voice-mail diary studies for naturalistic data capture under mobile conditions. In *Proceedings of Computer Supported Cooperative Work (CSCW)*, 87–95. New Orleans, Louisiana.

Perkins, R. 2001. Remote usability evaluation over the Internet. In *Design by People, for People: Essays on Usability,* ed. R. Branaghan, 153–62. Bloomingdale, IL: Usability Professionals Association.

Pickering, A., ed. 1980. *Science as Practice and Culture*. Chicago: University of Chicago Press.

Pine, J., and J. Gilmore. 1999. *The Experience Economy: Work is Theater and Every Business a Stage*. Cambridge, MA: Harvard Business School Press.

Polanyi, M. 1966. *The Tacit Dimension*. London: Routledge & Kegan Paul.

Pruitt, J., and T. Adlin. 2006. *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*. San Francisco, CA: Morgan Kaufmann Pub.

Pruitt, J., and J. Grudin. 2003. Personas: Practice and theory. In *Proceedings of Designing for User Experience*, 1–15. New York: ACM Press.

Randall, D., M. Rouncefield, and J. Hughes, J. 1995. Chalk and cheese: BPR and ethnomethodologically informed ethnography on CSCW. In *Proceedings E-CSCW*, 325–40. Stockholm, Sweden: ACM Press.

Rathje, W. L., and C. C. Murphy. 1991. *Rubbish! The Archaeology of Garbage*. New York: HarperCollins.

Rheingold, H. 2000. *Virtual Community: Homesteading on the Electronic Frontier*. Cambridge, MA: MIT Press.

Rich, M., S. Lamola, C. Amory, and L. Schneider. 2000. Asthma in life context: Video intervention/prevention assessment (VIA). *Pediatrics* 105:469–77.

Robinson, R. E. 1994. The origin of cool things. In *Proceedings of the American Center for Design Conference on Design that Packs a Wallop: Understanding the Power of Strategic Design*, 5–10. New York: American Center for Design.

Robinson, R. E., and J. P. Hackett. 1997. Creating the conditions of creativity. *Des Manage J* 8:10–6.

Rodden, T., and H. Anderson. 1994. Moving out from the control room: Ethnography in system design. In *Proceedings of the Conference on Computer Supported Cooperative Work*, ed. R. Furuta and C. Neuwirth, 429–39. New York: ACM Press.

Rogers, E. M. 2003. *Diffusion of innovations* (5th ed.). New York, NY: Free Press.

Rogers, Y., and V. Bellotti. 1997. How can ethnography help? *Interactions* 4:58–63.

Romney, A. K., W. H. Batchelder, and S. C. Weller. 1986. Culture as consensus: A theory of culture and informant accuracy. *Am Anthropol* 88:313–38.

Sachs, P. 1995. Transforming work: Collaboration, learning, and design. *Commun ACM* 38:36–44.

Said, E. 1978. *Orientalism*. New York: Pantheon.

Salant, P., and D. A. Dillman. 1994. *How to Conduct Your Own Survey*. New York: Wiley and Sons, Inc.

Schmidt, K., and L. Bannon. 1992. Taking CSCW seriously: Supporting articulation work. *Comput Support Coop Work* 1:7–40.

Schuler, D., and A. Namioka, eds. 1993. *Participatory Design: Principles and Practices*. Hilldale, NJ: Lawrence Erlbaum Associates.

Scola-Streckenbach, S. 2008. Experience-based Information: The role of Web-based Patient Networks in Consumer Health Information Services. *J Consum Health Internet* 12:216–36.

Scott, J. P. 2000. *Social Network Analysis: A Handbook*. 2nd ed. London: Sage Publication.

Shapiro, D. 1994. The limits of ethnography: Combining social sciences for CSCW. In *Proceeding of Computer Supported Cooperative Work*, 417–28. New York: ACM Press.

Shostack, G. L. 1984. Designing services that deliver. *Harv Bus Rev* 62:133–9.

Shostack, G. L. 1987. Service positioning through structural change. *J Mark* 59:34–43.

Shostack, G. L. 1993. How to design a service. *Eur J Mark* 16:49–63.

Smith, D. 1987. *The Everyday World as Problematic: A Feminist Sociology*. Boston, MA: Northwestern University Press.

Sonderegger, P., H. Manning, C. Charron, and S. Roshan. 2000. Scenario design. In Forrester Report, December 2000.

Squires, S., and B. Byrne, eds. 2002. *Creating Breakthrough Ideas: The Collaboration of Anthropologists and Designers in the Product Development Industry*. Westport, CT: Bergin & Garvey.

Suchman, L. 1983. Office procedures as practical action: Models of work and system design. *ACM Trans Office Inf Syst* 1:320–8.

Suchman, L. 1999. Embodied practices of engineering work [Special issue]. *Mind Cult Act* 7:4–18.

Suchman, L., J. Blomberg, and R. Trigg. 1999. Reconstructing technologies as social practice. *Am Sci* 43:392–408.

Suchman, L., and R. Trigg. 1991. Understanding practice: Video as a medium for reflection and design. In *Design at Work: Cooperative Design of Computer Systems,* ed. J. Greenbaum and M. Kyng, 65–89. Hillsdale, NJ: Lawrence Erlbaum Associates.

Suchman, L., R. Trigg, and J. Blomberg. 2002. Working artifacts: Ethnomethods of the prototype. *Br J Sociol* 53:163–79.

Thomke, S. 2003. R&D comes to services. *Harvard Business Review*, April, 71–9.

Trigg, R., J. Blomberg, and L. Suchman. 1999. Moving document collections online: The evolution of a shared repository. In *Proceeding of the European Conference on Computer-Supported Cooperative Work*, 331–50. Copenhagen, Norwell, MA: Kluwer Academic Publishers.

Wasson, C. 2000. Ethnography in the field of design. *Hum Organiz* 59:377–88.

Weller, S. C., and A. K. Romney. 1988. *Systematic data collection*. Newbury Park, CA: Sage.

Whiting, B., and J. Whiting. 1970. Methods for observing and recording behavior. In *Handbook of Method in Cultural Anthropology,* ed. R. Naroll and R. Cohen, 282–315. New York: Columbia University Press.

Whyte, W. F. 1960. Interviewing in field research. In *Human Organization Research,* ed. R. Adams and J. Preiss, 299–314. Homewood, IL: Dorsey.

Whyte, W. F. 1984. *Learning From the Field: A Guide From Experience*. Newbury Park, CA: Sage.

Wilson, S. M., and L. C. Peterson. 2002. The anthropology of online communities. *Annu Rev Anthropol* 31:449–67.

Wolf, M. 1992. *A Thrice-Told Tale: Feminism, Postmodernism, and Ethnographic Responsibility*. Stanford, CA: Stanford University Press.

Xun, J., and J. Reynolds. 2010. Applying netnography to market research: The case of the online forum. *J Targeting Meas Anal Mark* 181:17–31.

Yanagisako, S., and C. Delaney, eds. 1995. *Naturalizing Power: Essays in Feminist Cultural Analysis*. New York: Routledge.

Zhao, D., and M. Rosson. 2009. How and why people Twitter: The role that micro-blogging plays in informal communication at work. May 2009. In *Proceedings of the ACM 2009 Conference on Supporting Group Work*. New York: ACM Publishing.

Zimmerman, D. H., and D. L. Wieder. 1977. The diary: Diary-interview method. *Urban Life* 5:479–87.

Part VI

The Development Process

Section B: Design and Development

46 Putting Personas to Work

Employing User Personas to Focus Product Planning, Design, and Development

John Pruitt and Tamara Adlin

CONTENTS

Personas are a relatively young and popular technique used to help bring users into the forefront of people's minds during the difficult process of developing products and services. Yet, for many practitioners, the persona method remains largely ill-defined, haphazard, and mysterious. In this chapter, we present a simple framework for approaching the technique—the *persona lifecycle*—that sheds light on how personas fit into a standard development cycle. We then provide a little detail on two critical aspects of the method—creating and using personas. Our goal is to enable the reader to quickly and easily get started with the technique with a greater likelihood for success.

46.1 IT IS HARD TO BE USER CENTERED

In the best of all worlds, everyone working on a product would always be thinking of the needs of every person who will ever use the product. Real information about users would inform every decision and the resulting product would perfectly satisfy everyone who uses it. In practice, it is hard enough to get everyone working on a product to think about users at all (see Grudin [1990, 1993], for a discussion of obstacles and constraints in product development organizations). To deliver on the promise and benefits of user-centered design (UCD), we have to find creative ways to inject accurate information about real users into the chaotic world of product development.

It is a rare product indeed that does everything you want it to do in the way you want to do it. Why? Because, despite the fact that building products based on what real people need and want seems obvious, putting users (or rather, information about users) truly at the center of the design and development process is extremely difficult. Why is it so hard to be user centered?

First, being user centered is just not natural. Our more natural tendency is to be self centered—to design a product based on our own wants and needs (sometimes even when we are not an actual user of the product). Self-centered design is perhaps better than technology-centered design (another common inclination), but most of the time, the people on your product development team are not representative of the target audience for your product. Thus, as many practitioners in the UCD field have long evangelized (see, e.g., Nielsen [1993]), it is important to understand the user.

Second, users (really, people) are quite complicated and varied. It takes great effort to understand their needs, desires, preferences, and behaviors. And, pleasing some of them does not necessarily result in pleasing all of them.

Third, those doing the user and market research to understand your users (and others who are just more in touch with your users, e.g., the sales team and the support team) are not typically the people who design and build the product. Those who collect the data try to communicate the information it contains by creating presentations or reports, which are helpful only if they are absorbed by the right people at the right times. If the important information about users is not available at the right time, or is difficult to understand and remember, your development team will forge ahead to design and build the features they think their users would like.

46.1.1 The Word "User" Is Part of the Problem

When UCD was a new idea, simply injecting the word "user" into a design and development process was powerful: it challenged the status quo. Unfortunately, incorporating the word "user" into everyday corporate discourse is not enough to foster effective UCD.

Everyone (we hope) assumes that they are building products with users in mind; in many organizations, anyone asked would probably answer "yes, I think about the user a lot." However, people who talk about the "user" are almost never asked to further define the term and it is a sure bet that each person in the organization would describe the users in a different way. If everyone in the organization does not have a clear and consistent understanding of *who* they are building the product for, the product can fail. It is our contention that the word "user" cannot provide the clarity that is required.

> "User" is a catchall and ultimately a mean-nothing word. It reflects a technology-centric, rather than a people-centric view of the Web. To call someone a user is largely meaningless. The phrase "user-friendly" should never have had to be invented. It implies that technology is inherently hostile and that a new discipline—usability—had to be invented to make it friendlier. After all, we don't refer to cars as "driver-friendly."
>
> We don't refer to bicycles as "cyclist-friendly." We don't refer to chairs as "bum-friendly."
>
> **Gerry McGovern**
> *Consultant, gerrymcgovern.com.*
> *(From "Don't call people users." April 1, 2002.)*

Personas add the detail and specificity needed to provide product development teams the understanding needed to create user-centered products.

46.1.2 Personas Move Us beyond the Word "User"

Personas are fictitious, specific, and concrete representations of target users. Personas put a face on the user: a memorable, engaging, and actionable image to serve as a design target. The term "personas" was originally adopted and popularized by Alan Cooper in his 1999 book, *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How To Restore The Sanity* (see also, Cooper and Reimann [2003]).

Personas were born out of a short tradition in the UCD community toward user and system modeling and out of a somewhat longer tradition in marketing around market definition and customer representation. Perhaps the earliest was industrial designer Henry Dreyfuss who wrote about "Joe and Josephine," the heroes of his book *Designing for People* (1955), which were created to inform the physical design of objects and spaces. Geoffrey Moore, in his book *Crossing the Chasm* (1991), described the notion of "target customer characterizations"—images of customers, which are meant to replace or supplement impersonal and abstract market segments (see, e.g., Sissors [1966] or more recently, Weinstein [1998]). John Carroll (1995, 2000a, 2000b) has been a longstanding proponent of scenario-based design, which typically includes terse representations of users called actors or agents (see also Jacobson [1995]; Jacobson et al. [1992], regarding "actors" in "use cases"). Carroll argues that scenarios help designers and analysts focus on assumptions about people and tasks.

These and most other representations of users or customers (e.g., see Constantine and Lockwood [2001, 2002]; Mello [2003]; Upshaw [1995]) are typically *not* well-rounded or complete descriptions, but instead are confined to a few key details and specific contexts. Moreover, they just do not seem real, as they are devoid of life and personality. Personas, on the other hand, seem like real people. As such, personas carry information about users to your product team in way that other representations cannot. Personas enable us to move beyond our habit of referring to "users" and find a better way to communicate about and focus on the real people we want to use our products. Personas can humanize vast and disparate data sources, capitalize on our ability to remember details about individual people, and, in so doing, provide a usable and useful alternative to refer to "the user." Personas do the job of creating a concrete, focused, and stable definition on your audience. Based on our own experience with the approach, we believe that when created with data and used thoughtfully during the product development process, personas

- Make assumptions and knowledge about users explicitly, creating a common language to talk about users meaningfully
- Allow you to focus on and design for a small set of specific users (who are not necessarily like you), helping you make better decisions
- Engender interest and empathy toward users

These benefits as stated are really just a means to an end. Most importantly, as with all UCD techniques, the final goal is to create better designs, better products. We believe personas help accomplish this.

Support for these claims has been slow to accrue. The initial evidence was primarily based on a handful of case studies, anecdotes, and methods papers (see, e.g., Dantin [2005]; Freed [2004]; Grudin and Pruitt [2002]; Hourihan [2002]; Junior and Filgueiras [2005]; Kujala and Kauppinen [2004]; Levinson [2003]; Markensten and Artman [2004]; McQuaid, Goel, and McManus [2003]; Sinha [2003]; Shyba and Tam [2005]; see also the numerous sidebar stories in Pruitt and Adlin [2006]). Since the publication of our original book, more objective evidence in support of the method has begun to appear, including some experimental studies (e.g., Dharwada 2006; Long 2009). In Long's study, for example, groups of students were asked to solve a design problem in which some students used personas and others did not. Students using personas produced designs with better usability attributes. Their findings also suggested that personas can improve communication and facilitate more constructive discussion.

Additionally, there is a growing collection of conference presentations and peer-reviewed papers describing new case studies and extensions of the method (e.g., Antle 2006; Chang, Lim, and Stolterman 2008; Dharwada et al. 2007; Haikara 2007; Hill and Bartek 2007; Khalayli et al. 2007; Miaskiewicz, Sumner, and Kozar 2008; McGinn and Kotamraju 2008; Nieters, Ivaturi, and Ahmed 2007; Panke, Gaiser, and Werner 2007; Siegel 2010; Triantafyllakos, Palaigeorgiou, and Tsoukalas 2009; Tychsen and Canossa 2008). Most if not all of these report positive outcomes using personas or some general improvement to the approach.

Of course, the persona method has not been without critics (e.g., Chapman and Milham 2006; Chapman et al. 2008) and there are a few documented cases where personas did not result in beneficial effects (e.g., Blomquist and Arvola 2002; Rönkkö et al. 2004). Still, the lion's share of the evidence now points in favor of the method.

SIDEBAR 46.1 The Genesis of Personas in Product Design: Cooper Takes "Play Acting" One Step Further

Kim Goodwin
Vice President and General Manager, Cooper

With the publication of *The Inmates Are Running the Asylum* in 1998, Alan Cooper introduced the world to personas as a practical interaction design tool. However, Alan and the folks at his leading design consultancy, Cooper, had already been using personas for years.

In 1983, Alan was working as a solo software inventor. While working on a project management program he called "Plan*It" Alan realized he needed to understand more about how project managers thought, so he interviewed a handful of people. A woman named Kathy, who seemed the most typical, was the basis for the first persona-like model in Alan's head. While waiting for his program to compile, Alan would playact a project manager very much like Kathy, using the way she thought and worked to make decisions about the design of the application. Alan eventually sold Plan*It to Computer Associates, who sold it to the public as SuperProject. After that success, Alan went on to use this technique on other projects, including the visual programming language that became Visual Basic.

Later, when Alan ventured into consulting, he found that he could not just do what seemed right, because he first had to persuade other people. For this reason, on a 1995 project with Sagent Technologies, Alan created Chuck, Cynthia, and Rob—the first real, Goal-Directed personas. After some initial resistance, they worked as Alan had intended: they provided a way to keep everyone focused on what users really wanted to do, rather than on all the things they might do. (1) Since then, Cooper designers have refined and formalized the methods for researching, creating, and applying personas over the course of hundreds of projects. Many practitioners have begun using personas—some with excellent results, others not. The most common reason for failure? People miss the thing that makes personas so uniquely effective: they are based on a qualitative understanding of how real people behave and—equally important—why they behave that way.

Today's best personas are well researched, focused on behavior, and documented as a short-story writer would describe a beloved character with sympathy, respect, and just enough backstory to help you understand what makes them tick. (2) Effective personas are based on the kind of information you cannot get from demographics, survey data, or suppositions, but only from observing and interviewing individual people in their own environments. That qualitative, firsthand information is not only essential to design—it is essential to persuasion. If you cannot effectively persuade the programmers to build it, the executives to fund it, and the marketers and sales people to sell it, then the best design in the world is a failure. Personas are not just effective because they are accurate representations of human behavior. They are effective because they help both designers and stakeholders understand user needs at a gut level, which—in spite of all you hear about return on investment (ROI)—is where most business decisions are made.

46.2 MAKING PERSONAS PRACTICAL: THE CREATION OF THE PERSONA LIFECYCLE METHOD

If personas are such a good thing, why isn't everyone using them? Perhaps one answer is that creating and using personas is easier said than done. When we first started talking to persona practitioners in 2000, we noticed that many people in the UCD community could see the value of personas in their own work and to their organizations. However, those that tried to create and use personas were running into a fairly consistent set of problems in their persona efforts. While the idea of creating a set of target users is fairly straightforward, the actual process to create and use personas can be quite complex. We heard many of the same questions over and over again, such as the following:

- How do you decide whether personas are the right thing to do in your organization?
- How do you incorporate data into personas? What kinds of data work, and what kinds of data do not work?
- How do you know if personas are worth the effort it takes to create them?
- How do you communicate personas once they are created?
- How do you use personas to design great products?
- What do you do with personas once a project is finished? Can you reuse personas?

As we continued our research over the following years, we discovered that many practitioners were having less-than-stellar experiences with personas. Those that were able to create data-driven personas were finding that, if they are not well communicated and managed, even well-crafted personas are easy for designers and developers to ignore. At worst, poorly executed persona efforts yield no increase in user-focus and leach time and resources from other UCD techniques and other methods that can improve product quality. We found the following four common reasons for the failure of persona efforts:

- The effort was not accepted or supported by the leadership team.
- The personas were not credible and not associated with methodological rigor and/or data.
- The personas, and the techniques around using personas, were not well communicated.
- The product design and development team did not understand how to use the personas.

Once we fully understood the questions and common causes of failure, we focused our attention on finding solutions based on the input and insights of dozens of persona practitioners. The persona lifecycle was the result.

46.3 THE PERSONA LIFECYCLE

The persona lifecycle is a metaphorical framework that breaks down persona creation and use into sequential phases that map onto the life stages of human reproduction and development. There are five phases in this framework: (1) family planning, (2) conception and gestation, (3) birth and maturation, (4) adulthood, and (5) lifetime achievement and retirement (see Figure 46.1). The phases of the persona lifecycle framework bring structure to the potentially complicated process of persona creation and highlight critical (yet often overlooked or ignored) aspects of persona use.

As the name indicates, the persona lifecycle is a cyclical, mostly serial process model. As the illustration in Figure 46.1 shows, each stage builds on the next, culminating but not ending at adulthood. You will notice that the illustration also shows that final stage, lifetime achievement and retirement, is not immediately followed by the first stage. This is because different persona efforts culminate and restart in different ways; personas can be reused, reincarnated, or retired depending on the project.

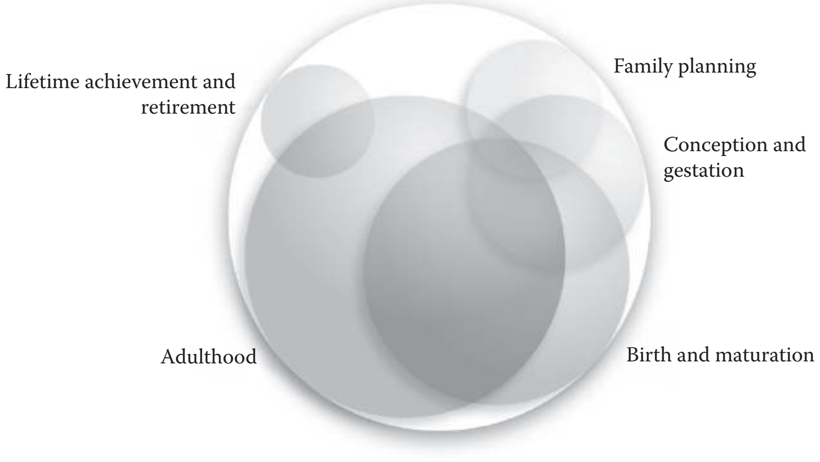


FIGURE 46.1 The five phases of the persona lifecycle. This diagram is designed to show both the order of the phases (from family planning through conception and gestation, birth and maturation, adulthood, and finally lifetime achievement and retirement) and the relative amount of effort and importance related to each phase.

If your company has already adopted some UCD methods, you should find that the phases of the persona lifecycle augment your existing process and help you get involved earlier in the product development cycle. Keep in mind that the persona lifecycle is not meant to replace other UCD tools and is not a complete user-centered product design method on its own. Rather, the persona lifecycle is an organized collection of processes and tools that will complement other familiar methods. You will use personas to enhance these other methods, particularly where there is a need for user definition and reference.

In the remainder of this chapter, we will discuss all of the phases of the persona lifecycle, focusing primarily on conception and gestation and adulthood because they are the least understood and, in many ways, the most critical phases for a successful persona effort. Conception and gestation and adulthood contain information on how to create personas from data (and/or assumptions, as necessary) and how to use completed personas in the planning, design, evaluation, and release of your products.

46.3.1 Phase 1: Persona Family Planning

Successful persona efforts are ones that are designed to solve specific problems for specific organizations and specific products. That is, personas created for one product cannot be easily adopted for another product. To be successful, you have to know what kinds of problems you want to solve and decide whether or not personas are the best way to solve them. Persona *family planning* is the strategy and planning phase that precedes the creation of the personas.

There are three major activities during the family planning phase:

1. *Researching your own organization* (which we call "organizational introspection")
2. *Creating a "persona core team"*
3. *Data collection*

46.3.1.1 Organizational Introspection

Before you begin any persona effort, you (perhaps with the help of the persona core team you assemble) should do some careful thinking about the particular problems you want to solve with your persona effort. While personas can help in many ways over the entire design and development cycle, they cannot solve every problem and they are not guaranteed to be accepted by the people you feel need them most.

Successful personas are those that meet the needs of *their* users and are built to fit seamlessly into their host environments. In the case of personas, the users are your colleagues, and the environment is your workplace with its existing design and development process.

Ironically, it is easy to forget to turn our analytic eyes on our users, the people on our teams and in our organizations who use the "products" we produce (e.g., research reports, storyboards, scenarios, prototypes, and other artifacts). We forget to carefully consider who our teammates are, their roles and responsibilities and goals, and what is working for them currently and what is not. We heap our user-focused processes onto teams who are interested and curious, but who, ultimately, just need to get their jobs done. As far as most product teams are concerned, they already know the fastest and most effective way to do their jobs; when push comes to shove and deadlines loom closer, your colleagues will inevitably revert to tried-and-true work habits.

Organizational introspection is, in simple terms, working to answer the following questions:

- What resources do we have for personas and other UCD activities?
- What product problems do we want to solve with personas?
- What process problems do we want to solve with personas?
- How can we ensure that the personas will be accepted and used by our colleagues?

Answering these questions now will allow you to decide whether personas will be appropriate and helpful. The answers will help you to create reasonable goals for the persona effort and predict the challenges you are likely to encounter as you create, introduce, facilitate use of, and maintain your personas. Armed with that information, you will create a plan for your persona effort that will target the application of your personas to appropriate aspects of your development process. Additionally, this information will help you determine what measures of success (and ROI) will be needed at the end of the project.

46.3.1.2 Creating a Persona Core Team

Personas are a simple idea, but they are inevitably more work than anyone expects. Successful persona efforts involve a core team of 2 and 10 people who are willing to find time in their calendars to dedicate to both persona creation and to helping evangelize the value and uses of the personas to the rest of the organization.

People you invite to join your core team do not necessarily have to understand personas to be helpful; people who are sensitive to the need for user focus in your company will make excellent core team members.

Your goal is *not* to create a team that will duplicate research or communication efforts; rather, it is to consolidate some aspects of these efforts such that they all contribute to the creation of personas. Plan to include the people who are already involved in user research, market research, business analysis, task analysis, or any other user- or customer-focused research or profiling activity. If you have colleagues in any of the following specialties, you should put them on the "short list" for inclusion on the core team:

- Information architects, interaction designers, and HCI specialists
- Usability specialists, user researchers, and ethnographers

- Technical writers and documentation specialists, training specialists
- Market researchers, business analysts, and product managers

These colleagues are likely to understand the value of personas, both for the organization and for their own projects. They bring with them a deep commitment to UCD, experience studying, analyzing, and designing solutions for target users, and an interest in new methods to bring user focus into the entire organization.

46.3.1.3 Identifying Data Sources and Collecting Data

From one-on-one interviews to widely published reports, there are thousands of data resources available to persona practitioners. Generally speaking, personas are best created from a combination of qualitative and quantitative research. The amount and types of data you will collect will depend on how much time and money you have to spend and your own evaluation of how much data will be necessary to create "good" personas for your project. If warranted and possible, you will do some original user research of your own, though this endeavor tends to be the most time consuming of all.

Most companies have a great deal of data "hanging around." Market research reports, customer segmentation studies, customer service logs, web usage logs and statistics—all of these sources can be reused in your persona effort. In fact, we recommend that you postpone future data collection efforts until after you analyze the data you already have during the conception and gestation phase of the persona lifecycle. This process will help you review your existing data sources from a new perspective, and you will be able to target future data collection projects to "fill in the holes" of your current understanding of your users. Look for both internal sources (like market research reports and interviews with product support specialists and other subject-matter experts) and external data sources (like public websites that provide statistical and demographic information).

As you identify and collect data sources, we also recommend that you identify and evaluate current assumptions about users that exist within your organization. The only assumptions that can harm your product are the ones you are not aware of. We strongly believe personas are much more credible and helpful if they incorporate and refer to real-world data. However, if data is simply unavailable, or you have no time to collect and analyze the data that is available, creating *assumption* or *ad hoc* personas is extremely worthwhile. One of the major organizational benefits of personas is their ability to focus everyone on a shared understanding of who the user is and what he or she really needs out of the product you are designing and building. The key word here is *shared*: everyone in your organization will inevitably build an internal understanding of the users of your product no matter what you do. If, through your efforts, they all picture the same users—even if this picture is built on assumptions—your product will benefit.

Family planning ends at the point at which you have established that personas are right for your organization and current project, you have buy-in from key individuals, you have a persona core team in place, you have a solid plan for the rest of the persona effort that suits your product team's needs, and your initial research and data gathering are complete.

SIDEBAR 46.2 Using Assumption Personas to Help a Multidisciplinary Team See the Need for Personas

Graham Jenkin
User-Centered Design & Research Executive,
Bank of America e-Commerce/ATM

In most companies, there are a range of team members that need to come together to define and build customer experiences: product managers, marketers, engineers, researchers, and user experience professionals. While the need for personas may be obvious to user experience professionals, other team members may require some creative persuasion.

At Bank of America, the e-Commerce User-Centered Design & Research (UCDR) team used "assumption" personas to obtain cross-team support for a dedicated persona development effort for its Online Banking product.

Online Banking team members from UCDR, Marketing, and Product Management were interviewed and asked to imagine one consistent customer profile and to make *assumptions* about that customer's relationship to the bank, usage patterns, motivations, goals, needs, frustrations, and attitudes toward technology.

The variation in the team members' assumptions was striking. Some assumed that Casey—the assumption persona: a young, urban professional—had strong loyalty to the Bank of America brand, while others assumed that she had no loyalty at all. Some assumed that Casey used Online Banking for day-to-day financial triage, while others assumed she used it for "big picture" planning. Some assumed that Casey was "geeky and wired," while others assumed she preferred to wait for family and friends to introduce her to new technology.

Of course, no one was right. Team member assumptions were exactly that—assumptions based on intuition and lacking in data.

When the interview results were shared, there was no doubt on the next action. Team members could not agree on who precisely Casey was, but they were unified in agreeing to develop and use a single set of personas based on real customer data.

46.3.2 Phase 2: Persona Conception and Gestation

Persona conception and gestation is the process of translating raw data into information and that information into personas. A lot of the work during the conception and gestation phase centers on collaboratively filtering data and organizing information—both information that arises out of the data you collect in family planning and information that arises from other sources, such as inherent knowledge of how people behave, your business or product strategy, the competitive marketplace, and technological affordances related to your product domain.

In this section, we will summarize the process we recommend for getting from raw data to completed personas. This process is designed to work best as a series of collaborative meetings with your core team. The first three steps cover persona conception; Steps 4 through 6 describe persona gestation.

46.3.2.1 Step 1: Identify and Assimilate Assumptions

Start with assumptions? After all this emphasis on data? You bet. Your completed personas will incorporate lots of data—but you will be introducing them into an organization that is overflowing with assumptions. If you can, identify categories of users that are important to your business and product domain.

How does your company refer to users today? Every company tends to already have a set of terms they use to describe the relationships and/or differences between their users. These terms are usually collections of defined *user roles, user goals*, and/or *user segments.*

Users defined in terms of segments (e.g., shared demographics or psychograpics):

- A large software manufacturer may think about their users as small businesses, medium businesses, large businesses, or home office users.
- A travel agency might be used to thinking about their users as *recreational travelers, families*, or *business travelers.*

Users defined in terms of roles (e.g., relationships they have with a system):

- A company that makes online presentation software might think about their users as *audience members, presenters, presentation designers*, and *technical support professionals.*

Users defined in terms of goals (e.g., what they are trying to achieve, in their own terms):

- An online bank might think that the most interesting differences between their customers are that they have different goals: *I have to feel my money is safe, I have to be able to access my account from anywhere*, or *I do not care about anything except low fees.*

The goal is simply to identify the ways people in your organization *already* talk about categories of users. Identifying these categories now (even if they are based solely on assumptions) will help you structure your data processing and build a bridge between the ways people think of users today and the data-driven personas you will create. When you are ready to communicate and use your personas, you will find it much easier to do so if you can describe them in language that is already familiar—even in the case where your data suggests that the initial categories should be replaced by different ones.

46.3.2.1.1 Get the Rest of the Assumptions out on the Table

Sit down with your core team and lots and lots of yellow sticky notes. Spend 30 minutes or so getting as many assumptions down on sticky notes as you can. Write one assumption on each sticky note. These assumptions should describe what you think your company's users are like as individuals. For the core team working at an online presentation, software company might create include the following:

- Forty-six-year-old sales guy who lives on the East Coast and has a hard time staying up-to-date with the latest product innovations taking place at the San Francisco home office.
- CEO who has to give a keynote presentation at a large conference, but he has to do it remotely.
- Investor relations specialist who has to present financial data to analysts and stakeholders at least once per quarter.
- Marketing manager who has to make sure that the key marketing messages for his or her company are distributed to the right people at the right times.

After everyone has created as many sticky notes as they can, it is time to find patterns in the assumptions. To do this, you will conduct an affinity exercise. Write the major categories of users (the ones you identified in Step 1) far apart from each other on a large sheet of paper. Ask everyone to place their sticky notes on the paper so that sticky notes with similar or related assumptions are near the appropriate category and near each other, and dissimilar assumptions are far apart. This exercise should be a fairly noisy, collaborative experience, as members of the core team discuss placement and groupings of their sticky notes.

If there are sticky notes that do not relate to any of the listed categories, create new areas on your large sheet of paper for them. As the exercise progresses, groupings of related sticky notes start to form "clusters." When the team feels that they are finished clustering, ask them to label the clusters. For example, one cluster could be labeled "Marketing people" and another, elsewhere on the big sheet of paper, could be labeled "Investor relations people."

Discuss what you have found as a group. Are there any surprises? It is a great time to think about the kinds of information you would like to find in your data, or collect directly from users. Your data will validate and enrich the categories or it will provide solid information to show that the existing categories are inappropriate. It will also allow you to define important subcategories of users that should also be expressed in personas.

46.3.2.2 Step 2: Process Your Data

It is time to process your raw data to extract information relevant to your user and product domains and then identify themes and relationships. While there are many ways you can go about processing your data to create personas, we strongly recommend that you conduct an affinity exercise like the one you have just done with assumptions. Before you get started, assign a number to each of your data sources.

46.3.2.2.1 Identify Key Data Points (Factoids) in the Data Sources

The first step in processing the data is to consume and filter the information that is in each of the research reports. You do this because not every data point in a given study/report is relevant to the definition of your target audience or the design of your product. Whether it is done before or during the meeting, ask your core team members to highlight findings that they think are key to understanding your target audience or that are highly insightful toward defining aspects of your product. In other words, you want them to look for findings that are relevant to your market, industry or domain. Highlight any facts that seem important to your product's audience or to the product itself.

Each important factoid should be copied or cut out of the original document. Remember to note the source and page number on each factoid so you can trace them back to the original sources later. We recommend that you use blue sticky notes for this exercise (signifying "cold, hard facts"). Whatever color you use, do not use the same color that you used for the assumption stickies; you will need to be able to distinguish between the assumptions and the factoids.

46.3.2.2.2 Assimilate the Factoids

Now the interaction (and fun) begins. To do the assimilation, everyone will get up and add their factoids to the clusters you created in the assumption exercise. Do not be afraid to move any of the stickies around as you do this—you can move assumptions, factoids, or even entire clusters as you assimilate.

As you continue assimilating, keep your eyes out for large clusters, or "puddles," of 8–10 or more sticky notes. Large clusters can usually be broken down further, and you should do this if you can. Usually, we recommend that everyone find spots for their stickies and then pair up to review the clusters together. During this "clean up" phase you will be able to break up puddles, add descriptive labels to clusters, and generally ensure that the clustering makes sense. Stop when the activity dies down and few stickies are still being moved.

Note that you will have clusters that contain both assumptions and factoids, but you will probably also find clusters of assumptions (with no factoids) and clusters of factoids (with no assumptions). This is illustrated in Figure 46.2. All three types of clusters are helpful. Clusters of assumptions without factoids will help you identify topic areas in which you need to collect more data. Clusters of factoids without assumptions tell you that there are aspects of your customers that you have not thought a lot about yet.

SIDEBAR 46.3 The Cooper Method: Collect Data Directly from Users and Identify Patterns to Create Personas

Kim Goodwin
Vice President and General Manager, Cooper

Some people have great success with personas, while others do not. Why is that? First, it is important to understand that even the best personas will not solve all of your problems—scenarios, design judgment, and visualization skill are equally important. Assuming you have all of those things, the key to success with personas is to do the right kind of research and to make sure your personas truly reflect your findings.

1. START WITH THE RIGHT KIND OF RESEARCH

"Personas" that are made up without data are not really personas, and although they can still be useful thought exercises, they are far less effective than real personas that are based primarily on ethnographic user data. Ethnographic techniques are valuable because they assume that an interview subject's attitudes and behaviors are so habitual as to be unconscious. Rather than asking users what they want, it is more effective to focus on what users do, what frustrates them, and what gives them satisfaction. By combining interviewing with direct observation—preferably in the actual usage context—you can get a lot of data very quickly. Observation also helps minimize dependence on users' self-reported behavior, which is often inaccurate.

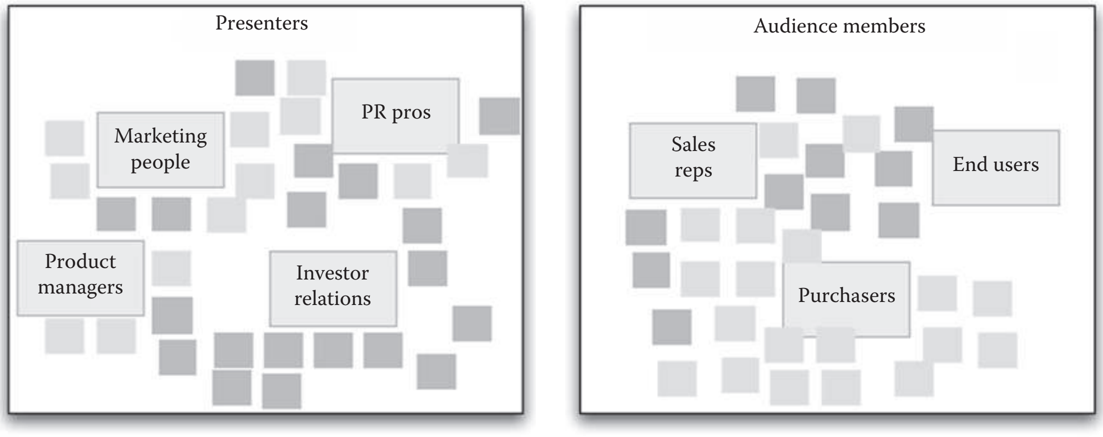


FIGURE 46.2 An example of clustered assumptions (lighter squares—"yellow" stickies) and factoids (darker squares—"blue" stickies) with cluster labels (larger squares—"pink" stickies) and initial categories of users.

At Cooper, we send the designers out to do the research, so they see the problems firsthand and develop empathy with the users. We may occasionally have three interviewers, but two of them are consistent across all of the interviews; this makes synthesis much easier later on. We spend 45 minutes to an hour with individual people. It would be easy to write a whole chapter on research techniques alone, but there are a few fundamental points. First, ask a very broad question (such as "Could you think of a typical work day recently, and walk me through it?") This raises a number of issues the interviewers can pursue without needing to ask leading questions. Another important technique is to stay case-focused; in other words, ask for specific instances rather than generalizations. This will get you more detailed and more accurate information. Also, be sure to look at and ask questions about artifacts or aspects of the environment. Finally, focus on actual behavior and frustrations, rather than asking the users to design the product for you.

Contrary to some expectations, this kind of research can be done in anywhere from a few days to a few weeks. You will know you can stop interviewing when you can predict how each user will respond; this means patterns are beginning to emerge. If you have the time and budget, you can verify your findings with quantitative surveys or other techniques, but these cannot replace direct observation.

2. IDENTIFY BEHAVIORAL PATTERNS FROM THE DATA

Once you finish interviewing, list all of the behavioral variables for each user role, that is, ways in which interviewee behavior differed. In an online shopping domain, for example, you might have variables such as frequency of shopping, degree of enjoyment, and price versus service orientation. There may also be demographic variables that seem to affect behavior, such as age or technical skill. Be wary of focusing on demographics during persona creation, since behavioral variables will have far more impact on the design. Note that if you are doing an enterprise application, each role will have its own set of behavioral and demographic variables. Although the number of variables will differ from project to project, it is typical to find 20 or so variables per role. Map each interviewee against the appropriate set of variables, then look for people who clump together across a large number of the variables. When you have found a set of people clustering across six or eight variables, there is a good chance that you have found a major behavior pattern that will form the basis of a persona.

3. TURN YOUR PATTERNS INTO PERSONA DESCRIPTIONS

For each pattern, add details based on your data. Describe the current potential usage environment, typical workday (or other relevant time period), current solutions and frustrations, relevant relationships with others, and goals. Avoid the temptation to add a lot of irrelevant personal detail; if you are designing an e-mail tool, it does not matter that your persona wants to be an actress. One or two bits of personality can bring an otherwise dull persona to life, but too much biography will be distracting and will make the persona less credible as an analytical tool. If every aspect of the description cannot be tied back to real data, it is not a persona—it is a creative writing project that should not be used for making critical design and business decisions. Describe each persona in a one- or two-page narrative that helps stakeholders understand what makes them tick.

4. USE THE PERSONAS TO DRIVE SCENARIOS, REQUIREMENTS, DESIGN … AND COMMUNICATION

Use your personas to develop scenarios; put the personas in realistic future situations and envision how they would like a magic black box of a product to work. These scenarios will lead you to a set of needs you can discuss with stakeholders. You can describe requirements from the personas' point of view, which leads to less resistance than requirements that come from you. Once there is agreement on the requirements, additional scenarios (along with good design skills!) will help create the conceptual framework for the design. When you illustrate the design direction in a scenario, stakeholders are more likely to see the value of the solution. Personas will help you all the way to pixels and specifications, and even through implementation.

46.3.2.3 Step 3: Identify Subcategories of Users and Create Skeletons

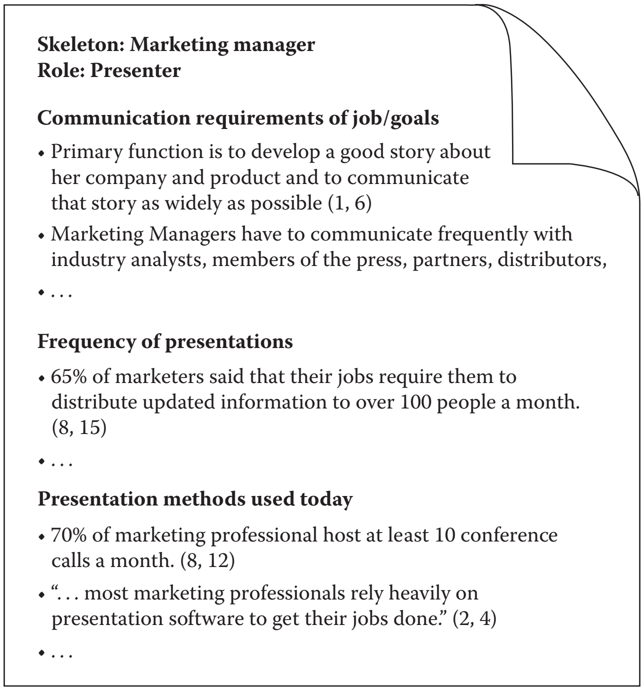
Look at the data clustered under each of your user categories. As a team, evaluate and discuss the possibility that each category should be divided into two or more subcategories. Consider roles, goals, and segments in this assessment. As you identify subcategories, you can write them on a whiteboard and you may also find it helpful to transfer the subcategory names onto sticky notes and place them appropriately in your assimilated data. In doing this exercise, you are simply exploring the possible groups of users that have emerged from your data. Try to identify "differences that make a difference" within each category, based both on the clustered assumptions and the clustered factoids.

46.3.2.3.1 Create Skeletons

Once you have identified and agreed upon the categories and subcategories of users, you are ready to create skeletons. Skeletons are very brief, usually bulleted lists of distinguishing data ranges for each subcategory of user. Skeletons help your core team transition from thinking about categories of users to focusing on specific details; they also allow your team to present the key findings of the assimilation exercise to stakeholders.

Create one skeleton for each of the subcategories you identified. On each skeleton, list the cluster labels that relate to that subcategory; these cluster labels will become headings in your skeleton (see Figure 46.3). Because you will be comparing and prioritizing skeletons against each other, it is important that each one contain at least somewhat comparable information. Consider including common characteristics or headings across all of your skeletons. (If you do this, you may find that you are missing information for some skeletons. In those cases, either leave that information blank, perhaps marking it as "need data," or make an informed estimation about what it might be. If you do the latter, be sure to indicate that it is an assumption to be followed up on.)

Feel free to create as many skeletons as you and your team feel are necessary to "cover" the discoveries you made during the clustering processes. You will have the opportunity to combine and prioritize them in the next steps.



Skeleton: Marketing manager
Role: Presenter

Communication requirements of job/goals

- Primary function is to develop a good story about her company and product and to communicate that story as widely as possible (1, 6)
- Marketing Managers have to communicate frequently with industry analysts, members of the press, partners, distributors,
- . . .

Frequency of presentations

- 65% of marketers said that their jobs require them to distribute updated information to over 100 people a month. (8, 15)
- . . .

Presentation methods used today

- 70% of marketing professional host at least 10 conference calls a month. (8, 12)
- ". . . most marketing professionals rely heavily on presentation software to get their jobs done." (2, 4)
- . . .

FIGURE 46.3 An example of a skeleton created out of assimilated data. Note that the skeleton includes factoids grouped according to topic; it does *not* include any narrative details or "personalized" information.

SIDEBAR 46.4 Sometimes Categories of Users (and Personas) Should Be Based on Verbs, Not Nouns

Karen McGrane
Vice President, User Experience, Avenue A | Razorfish

Though the method of developing personas is relatively new, it seems a given that a persona maps to, well, a person. The approach taken by user-centered designers focuses on understanding the needs, goals, and mindsets of a given individual. A persona usually describes a single, stable individual, whose interaction with a company, product, or website stays consistent throughout the experience.

What happens when the goals and mindsets of an individual user can change rapidly over the course of their relationship with a company—or even over the course of a single session? When we observed people visiting the *New York Times* website, we realized that individual users did not fall neatly into a single mode of use. People tended to switch their goals over time. People used the site differently in the morning, when they wanted to read headlines and catch up on the day's news, as compared to the afternoon, when they might be looking to take a break over lunch or between meetings. We often observed people change modes in the course of a single session, moving fluidly from a news-reading mode, scanning headlines, into a planning mode, where they would research restaurants and read movie reviews.

When it came time to document our personas, we realized that the standard way of documenting them would be limiting. We did not want to imply that "Ketan, a 54-year-old technology consultant" was our only "news junkie," or that "Lisa, a 32-year-old account executive" would always operate in a "planning mode." Making a one-to-one connection between the person and the action seemed inaccurate, since we knew each person would most certainly take multiple actions during their use of the site.

What's more, putting the emphasis on the noun (the persona) rather than the verb (the activity) did not really help the *New York Times* achieve their business goals. While they absolutely wanted to create a usable experience for a visitor operating within a given mode, they also wanted to know what would prompt someone to change modes—since shifting modes extends the experience to more page views and more frequent visits. As experience designers, we concluded that we did not need to focus on the person(a). We needed to focus on the activity—the verb, the action, the mode of use (Figure 46.4).

Clients are eager to understand their customers better, and many focus on personas as the tool that will help them do that. In the future, I will encourage my clients to think about other means of understanding customer behavior. I think one of the most important ways they can do this—outside of personas—is to learn more about customer goals and modes of use.

46.3.2.4 Step 4: Prioritize the Skeletons

Once you have a set of skeletons, it is time to review them in preparation to get feedback from all stakeholders. When you meet with stakeholders, you will evaluate the importance of each skeleton to your business and product strategy and prioritize the skeletons accordingly. The stakeholders will help to identify a subset of skeletons to develop into personas.

Before you meet with the stakeholders, it is a good idea to do some initial prioritization of the skeletons. The core team should carefully review the skeletons you have created and make sure you agree that each one truly does reflect a "difference that makes a difference" between subcategories of users as identified earlier. In many cases, it will be possible to reduce the number of skeletons by combining several of them. Once you have a set that you feel should not be further reduced, prioritize them in a way that makes sense to the core team. This will give the stakeholders as "strawman" to work with—and that is easier than asking them to prioritize a series of skeletons from scratch. As you do this, think about the following:

- Is this category or subcategory important to our product (relevant, unique, illuminating)?
- How important is it to our business?
- Are there any groups missing?
- Are some of the categories almost right, but a few of the characteristics are "off" and need to be tweaked?

Note that you should prioritize skeletons *within* each major category of users. For example, prioritize all the "business traveler" skeletons relative to one another, and the "families" skeletons separately. Eventually, you will want to know the priorities of your personas *across* these categories, but first you will need to narrow in on the correct skeletons *within* each category.

Now schedule a meeting with stakeholders empowered to make decisions about the strategic focus of the company. If stakeholders are not aware of the data and general process that led to these skeletons, present that information before

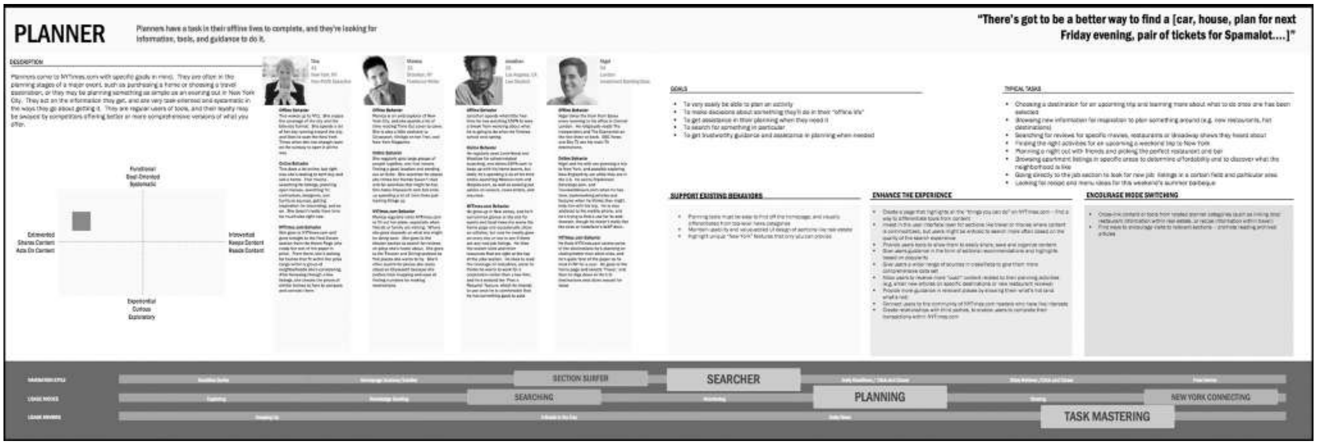


FIGURE 46.4 One of the "verb-based" persona posters for the *New York Times* project. Note that the poster is focused on the word "Planner" and not on a specific persona name. In fact, the "Planner" poster includes details about four individual personas.

introducing the skeletons to them. Introduce the skeletons in one category at a time, and ask them to assign priorities based on the following:

- Frequency of use: How often would each skeleton use your product? Daily users would likely be more important regarding design decisions than those that only use your product once a month.
- Size of market: Roughly how many people does each skeleton represent? Larger markets are usually more important than smaller ones. Do you plan to aim your new product at a new market? In that case, you might consider the importance of a small market with growth potential.
- Historic or potential revenue: How much purchasing power does each skeleton encompass? If this is a new product, you may have to estimate this amount (e.g., through trade journals, market trends, market research, understanding spending behaviors in different by related markets). In many cases, users might not directly make the purchase; someone else buys such products for them. Still, they may influence those purchase decisions.
- Strategic importance: Decide who is your most strategically important audience. Is it those who make the most support calls, those that rely on your product for critical activities, those that use your competitor's product, those that do not use yours or anyone's product yet? Are you trying to expand or grow your market? If that is your primary goal, do your skeletons include nonusers, technology pioneers, or trend setters? Which target audiences will help your team innovate or stretch?

Prioritization can be difficult, and your first meetings will likely end in requests for more data. For example, stakeholders might ask you to find data on the market size for each of the skeletons before they feel comfortable prioritizing. If prioritization feels impossible, it might mean that the business goals are not clear. This can lead to some delicate conversations. In general, remind the stakeholders that you want to prioritize the skeletons so that they accurately reflect the company's vision for itself and the product. The stakeholders *must* be involved in the prioritization of the skeletons, because they are the ones who are setting the course for the business. Be willing to go back and try to find more data if necessary.

46.3.2.5 Step 5: Develop Selected Skeletons into Personas

You now have a reduced set of basic skeletons. Your task at this point is to enrich these skeletons to become personas by adding additional data as well as concrete and individualized details to give them personality and context. You will also include some storytelling elements and photos to make them come to life.

As you build on your skeletons, all the details of your personas will be encapsulated in a "foundation" document. Foundation documents contain the complete definition of a given persona, but they do not have to be long or difficult to create. Depending on your goals and the needs of your team, your foundation document could range from a single page to a long document. Creating a foundation document for each persona will provide you and your team with a single resource you can harvest as necessary as you create your persona communication materials. At the very least, complete personas must include core information essential to defining the persona and the goals, roles, behaviors, segment, environment, and typical activities that make the persona solid, rich, unique, and more importantly, relevant to the design of your product.

If you are extremely time and resource constrained, you can start with brief one-page or resume-style foundation documents. Then, as you find the time, you can always come back and add to the information in these short foundation documents.

When you are deciding which characteristics to include in your foundation documents, think about the types of information that will be most helpful to your core team and to the development team. We recommend you include at least rudimentary information in each of the following categories:

- Photo(s)
- Name
- Age
- Personal details/family life
- Income/spending habits
- Work/job details
- Use environment/artifacts
- Activities/use scenario
- Knowledge/skills/abilities
- Goals/motives/concerns
- Likes/dislikes
- Quotes
- Market size/influence

To create the content of these sections, you will turn the characteristics in your skeleton personas into very specific, concrete values. These specific details turn your collection of facts into representations of people. For example:

- High-Tech, publicly traded companies *becomes* "PrinterCo"
- Works in a communications role *becomes* Marketing Manager
- Seventy percent female *becomes* Megan, Dianne, Irene, and so on.
- Lives in a major metropolitan city *becomes* Chicago, LA, or Houston

As you replace factoids with specific details to enrich your persona, copy the factoid or set of factoids into a comment or a footnote in your foundation document. A lofty but worthy goal is to have every statement in your foundation document supported by user data. You likely will not achieve this, but the attempt helps you to think critically about your details, and highlights places where you might want to do further research.

46.3.2.5.1 *Moving toward Precision Means Moving away from Accuracy*

In many cases, the accuracy of your data lies in its ranges (not just central tendencies, but descriptors of variance, percentages and skew), and by selecting precise descriptors, you are going to lose some of that accuracy. For example, if a category includes males and females, you cannot create a single individual who "represents" the entire category. Rather than trying to represent every nuance of the entire category, try to pick values that are reasonable, believable and meaningful.

Think of your data, and your categories and subcategories of users, as describing *neighborhoods* of related users of your product. As you create your personas, you are describing a single, specific "resident" of each neighborhood; as in real life, each resident inhabits his or her neighborhood, but no one resident can represent all the qualities of all the people in the neighborhood.

46.3.2.5.2 *Incorporate Narrative and Storytelling Elements*

Enriching your terse skeletons into personas that are realistic and engaging requires some storytelling. To do this well, consider that you are trying to "tell the story" of the data in your foundation documents with narrative. What do your personas sound like and act like? What can they do or not do? Turn your factoids and specific details into a running story; a sequence of actions and events with interaction and even a plot. Demonstrate their interactions with people, objects, and systems. In the best case, these stories are modeled on real, representative cases revealed through qualitative research.

Narratives in persona documents usually are written in third person, active voice. The following is an example of a beginning descriptive overview for a marketing manager named Megan written as a narrative:

> Megan is the product marketing manager for PrinterCo, a leading printer manufacturer. Part of Megan's job is to create and foster the company's image as a cutting-edge technology company. Her primary function is to develop a good brand about her company and product and to communicate that brand as widely as possible. She spends her time thinking about how to educate people about PrinterCo's products and to build relationships between customers and PrinterCo.

Be careful when evoking stereotypes or any information that could elicit a strong personal response. When in doubt, choose details that help others see your persona as a real person, with particular goals, needs, and interests that are understandable. Allow realism to win out over political correctness; avoid casting strongly against expectations if it will undermine credibility. Break the mold if it helps get people on board with your effort. Alan Cooper addresses this issue by stating: "all things being equal, I will use people of different races, genders, nationalities, and colors." (p. 128)

46.3.2.5.3 *Illustrate Your Personas*

Each persona needs a face, a photo or set of photos, to make them real. We believe real photos or illustrations are critical; they help your team believe in the personas and understand that each persona describes a single person. The choice of what specific photos to use is a hard one. These illustrations of your personas are extremely influential, and can significantly affect how your personas are perceived.

A photo is more than just a face; the model's clothing, expression, activity, and general appearance along with the setting and background will all communicate or even dictate some of the characteristics of your persona. You can either take advantage of this fact or continually fight it. In general:

- Avoid "slick" stock photos. Photos of models *look* like photos of models—not of real, everyday people. A great alternative is to take photos of friends-of-friends. Do not use photos of a person that anyone on the product team knows. If you do take your own photos, take many! You can use different photos at different times to keep people interested in the personas.

- If you cannot take your own photos, look for photos online. Flickr.com is a great example of an online photo sharing site, and you can look in the Flickr “Creative Commons” area for work by photographers who are willing to let others use their photos royalty-free. Do a search for “portraits” to find hundreds of potential personas!

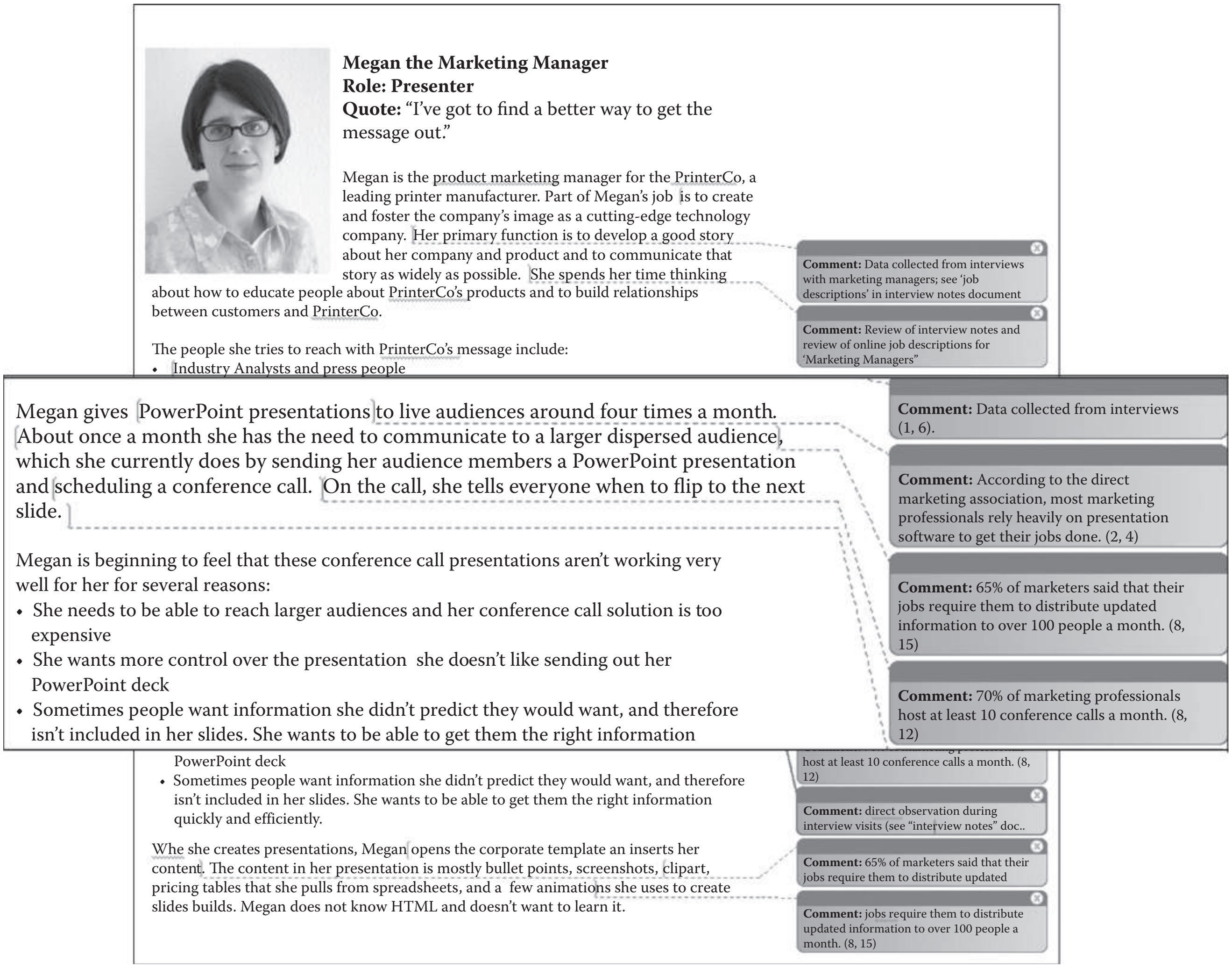
If possible, include multiple photos in your foundation documents to illustrate your persona (Figure 46.5).

46.3.2.6 Step 6: Validate Your Personas

Once you have added details, it is important to double-check to make sure your final personas still reflect your data. Your goal is to ensure that you did not stray too far away from your data when you made their characteristics specific and concrete and added elements of story telling. While it is true that personas cannot and do not need to be completely accurate, you do want to ensure that they reflect the essential information about your target users that you found in your data. If you built assumption personas, you want to ensure that the personas you created really do capture the assumptions in your organization.

To validate your personas, you can do one or more of the following:

- Review your personas against the original data sources
- Have experts, those who are closest to your users, review your personas
- Have representative users of each persona review “their” personas
- Conduct “reality check” site visits with real people that loosely fit into each persona’s category or role
- Conduct large sample surveys or interviews and apply statistical analysis or modeling



Megan the Marketing Manager
Role: Presenter
Quote: “I’ve got to find a better way to get the message out.”

Megan is the product marketing manager for the PrinterCo, a leading printer manufacturer. Part of Megan’s job is to create and foster the company’s image as a cutting-edge technology company. Her primary function is to develop a good story about her company and product and to communicate that story as widely as possible. She spends her time thinking about how to educate people about PrinterCo’s products and to build relationships between customers and PrinterCo.

Comment: Data collected from interviews with marketing managers; see ‘job descriptions’ in interview notes document

Comment: Review of interview notes and review of online job descriptions for ‘Marketing Managers”

The people she tries to reach with PrinterCo’s message include:

- Industry Analysts and press people

Megan gives PowerPoint presentations to live audiences around four times a month. About once a month she has the need to communicate to a larger dispersed audience, which she currently does by sending her audience members a PowerPoint presentation and scheduling a conference call. On the call, she tells everyone when to flip to the next slide.

Comment: Data collected from interviews (1, 6).

Comment: According to the direct marketing association, most marketing professionals rely heavily on presentation software to get their jobs done. (2, 4)

Comment: 65% of marketers said that their jobs require them to distribute updated information to over 100 people a month. (8, 15)

Comment: 70% of marketing professionals host at least 10 conference calls a month. (8, 12)

Megan is beginning to feel that these conference call presentations aren’t working very well for her for several reasons:

- She needs to be able to reach larger audiences and her conference call solution is too expensive
- She wants more control over the presentation she doesn’t like sending out her PowerPoint deck
- Sometimes people want information she didn’t predict they would want, and therefore isn’t included in her slides. She wants to be able to get them the right information

PowerPoint deck

- Sometimes people want information she didn’t predict they would want, and therefore isn’t included in her slides. She wants to be able to get them the right information quickly and efficiently.

host at least 10 conference calls a month. (8, 12)

Comment: direct observation during interview visits (see “interview notes” doc..

Whe she creates presentations, Megan opens the corporate template an inserts her content. The content in her presentation is mostly bullet points, screenshots, clipart, pricing tables that she pulls from spreadsheets, and a few animations she uses to create slides builds. Megan does not know HTML and doesn’t want to learn it.

Comment: 65% of marketers said that their jobs require them to distribute updated

Comment: jobs require them to distribute updated information to over 100 people a month. (8, 15)

FIGURE 46.5 A portion of an example persona foundation document. Note that the callouts on the side of each page are “factoid” references.

These five approaches are not mutually exclusive, nor are they the only means to validate your personas. Treat the validation process as an opportunity to gather even *more* data about your users, incorporating significant findings back into the persona definitions.

46.3.2.6.1 Completed Personas Do Not Mark the End of User Research

At the point that you finish the creation of your personas, you may be tempted to think that you do not need to further understand (do research) or involve real users in the development of your product. From our perspective, this could not be further from the truth. We believe personas are a great starting point for understanding and incorporating user information in the development cycle.

46.3.3 Phase 3: Persona Birth and Maturation

Personas are typically created (contained) and communicated as static documents or posters that provide a snapshot of interesting and relevant information about users. In fact, the foundation document we describe above is a case in point. These artifacts have proven helpful, largely because they help make information about users highly accessible, engaging, and memorable to people making decisions. However, such representations of users are not "alive." They are depicted as motionless portraits, usually contained within a single, finite document and presented as such. There is no room for growth or development. That is, unlike a character in a book or film, such descriptions do not evolve. Moreover, the team using them is supposed to "get to know them" almost instantly. When we get to know a friend, neighbor, colleague, or even a character in a favorite book or TV show, we *build up* an understanding of them, a relationship with them. Once we know people, we are able to understand why they do what they do, what they want, and what they need. Engendering this level of understanding is the next frontier for user representation.

We believe you have to enable the personas to "come to life," allowing them to be alive and to develop in the minds of the people using them. Toward this end, we propose that persona practitioners must do the following:

- Embrace the challenge of communicating information about users through narrative and storytelling
- Incorporate a variety of formats and media to communicate the essential persona characteristics
- Maintain a lifecycle perspective when educating colleagues about the personas
- Allow the people using the personas to extrapolate from and extend them

In other words, personas should be more than a static collection of facts. Personas are compelling stories that unfold over time. To be very clear, we are NOT suggesting that personas change drastically over time, take on new characteristics, or develop new skills; they are not to be moving targets. We believe that successful personas and persona efforts are built progressively; just like we get to know people in our lives, we must get to know personas (and the data they contain) by developing a relationship with them. No single document, read in a few minutes or posted on a wall, can promote the kind of rich, evolving relationship with information about users that is the cornerstone of good product development. No single document can contain the wave of scenarios and stories that your personas will inspire. Personas must be aided, if they are to *live in the minds of your colleagues*. As long as the personas are well-built, data-driven, and thoughtfully communicated, the product team can use the personas that come to exist to generate new insights and seek out the right details when they need them.

Like parents sending young children off to school, you and your core team will send your personas into your organization to interact with other people. The personas are fully formed but may continue to evolve as your team becomes familiar with them. Problems at this phase can lead to a lack of acceptance or visibility or personas that "die on the vine" and disappear from the project. More subtly, your personas may come to be misconstrued and misinterpreted. Successful persona birth and maturation requires a strong, clear focus on communication to ensure that your personas are not just known and understood, but adopted, remembered, and used by the product team. The birth and maturation process includes the following:

- Creating a persona campaign plan to organize your work in birth and maturation and adulthood
- Introducing the personas (and the persona method) to the product team
- Ensuring that the personas are understood, revered, and likely to be used, (e.g., creating artifacts to progressively disclose persona details)
- Managing the minor changes to the persona descriptions that become necessary after the personas are introduced

At this phase, you must be prepared to answer the difficult questions that will inevitably come up as you introduce the personas; you will have to be prepared to discuss the process you used to create the personas, their utility, the ways you would like the product team to use the personas, and the ways you intend to measure the value of the persona effort. The work you did in the family planning phase will come in quite handy as you prepare your answers to these inevitable questions.

SIDEBAR 46.5 Using Personas to Build Teams

Aviva Rosenstein

Manager, Design Research, Yahoo! Media Group

The Yahoo Photos team supports an application for sharing, printing and organizing personal photos. This team has been in the vanguard in adopting Agile development approaches at Yahoo as an alternative to the traditional waterfall development process. To make this transition to an Agile approach successful, we needed to find ways to bring engineers and designers together into a single, cohesive team.

When introducing the persona set to the project team, I used our design personas to create connections between the engineering and design staff while, at the same time, familiarizing the entire team with our target customers, by creating a simple game requiring knowledge of each of the personas' characteristics. The design team scheduled a happy hour to introduce the personas, created handouts and posters with information about each persona, and invited the entire product team, including all the designers, developers, and product managers assigned to the property to come play the game.

The exercise was a success; everyone had to study the persona artifacts to determine what questions they could ask, which gave them a head start in internalizing our personas. In addition, the game provided an enjoyable context for designers and engineers to cross social barriers and begin interacting with each other.

The persona artifacts served as "boundary objects" (Star and Griesemer 1989): a way of bridging communication gaps between disparate functional and organizational worlds. The exercise itself broke down the barriers that existed between the engineering and design groups and allowed the design team to get the rest of the developers to empathize with the needs of our customers, but the actual impact of the persona artifacts went even further. After this exercise, team members commonly and naturally began to use our personas to refer to actual customers and their needs, in storyboarding, use cases and requirements documentation. Project management used the personas to communicate value propositions with executive staff. Marketing used the persona artifacts to communicate target audience characteristics to our outside public relations (PR) agency. Design management used the handouts to communicate requirements to an outside design firm retained to develop a specific part of the service. It gave all of the teams a shared language for talking about our business. In addition, introducing the persona set in an informal and engaging way encouraged adoption among the various disciplines contributing to the overall success of the project.

46.3.3.1 Persona Artifacts

You can use a variety of methods to communicate personas to the members of your product team, including websites, posters, illustrations, Word documents, Visio diagrams, live actors, and videos. Remember that the artifacts and materials you create to communicate information about personas are very important—they are the user interface (UI) for your personas and the data behind them. Well-thought-out and well-designed persona materials can add credibility to your entire persona effort and help enormously with your persona communication campaign (Figure 46.6).

We define three major categories of persona artifacts:

- *Buzz generators.* These artifacts should be designed to build up anticipation about the introduction of the personas. They are usually posters with relatively little information on them. Buzz generators can give hints about the fact that personas are coming (e.g., a poster that says "do you know who your users are?") or begin to introduce the personas themselves (e.g., a wanted-style poster showing just a persona's photo, name, and perhaps role: "Meet Barry the Business Traveler").
- *Comparison facilitators.* Comparison facilitators are helpful after everyone has been introduced to the personas. These artifacts should be designed to help people understand key differences between the personas. For example, you could create a table-style poster that lists the personas across the top and highlights different goals, technical abilities, challenges, and so on across each row.
- *Enrichers.* As the design and development process continues, you can use enricher artifacts to refresh everyone on specific aspects of each persona. For example, when the team begins working on security features, you could create enricher artifacts that describe the security challenges for each of the

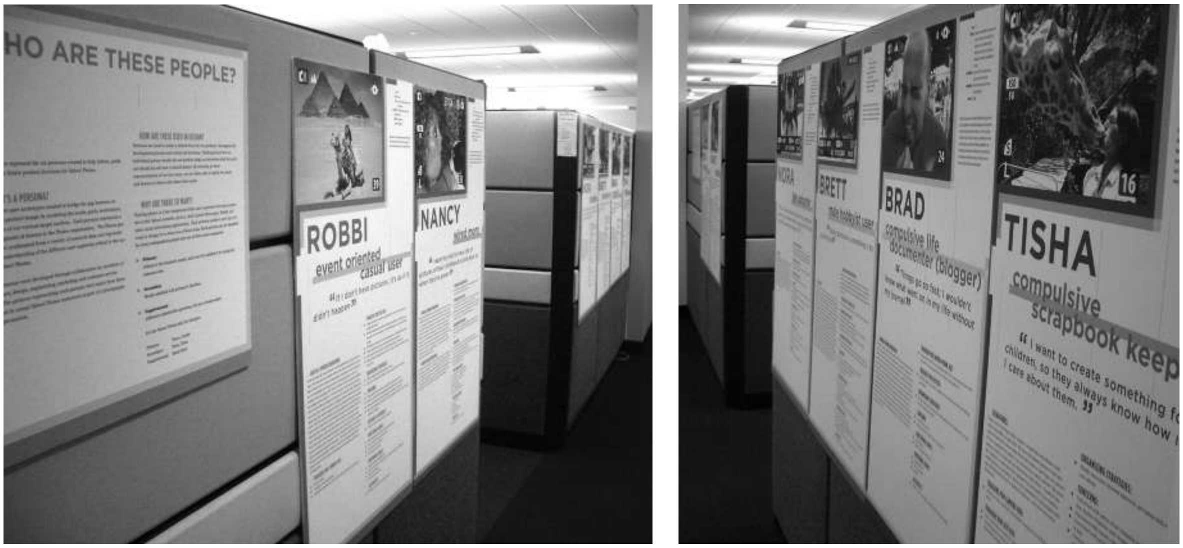

HO ARE THESE PEOPLE?
ROBBI
event oriented casual user
NANCY
NORA
BRETT
BRAD
compulsive life documenter (blogger)
TISHA
compulsive scrapbook keep
I want to create something f children, so they always know how I care about them.

FIGURE 46.6 The Yahoo! Photo persona artifacts helped bridge a communication gap between functional and organizational worlds.

> personas. You might also send out a monthly e-mail "from" each persona to update the team on their goals and needs. You can also create fun artifacts as enrichers; for example, you can hand out candy bars that have new "persona" labels listing salient facts about each persona. Anything that helps keep the personas fresh and alive in the minds of the team is a good thing!

Note that it does not take a lot of these artifacts to have an effective communication campaign. Be very strategic and frugal in your choices. Approach the creation and distribution of your persona artifacts carefully.

46.3.3.1.1 Agree on the Specific Goal of the Artifact

Why are you creating this specific artifact? The goal will probably be related to one of the three categories of artifacts we described earlier.

46.3.3.1.2 Agree on the Audience, Timing, and Distribution Method for the Artifact

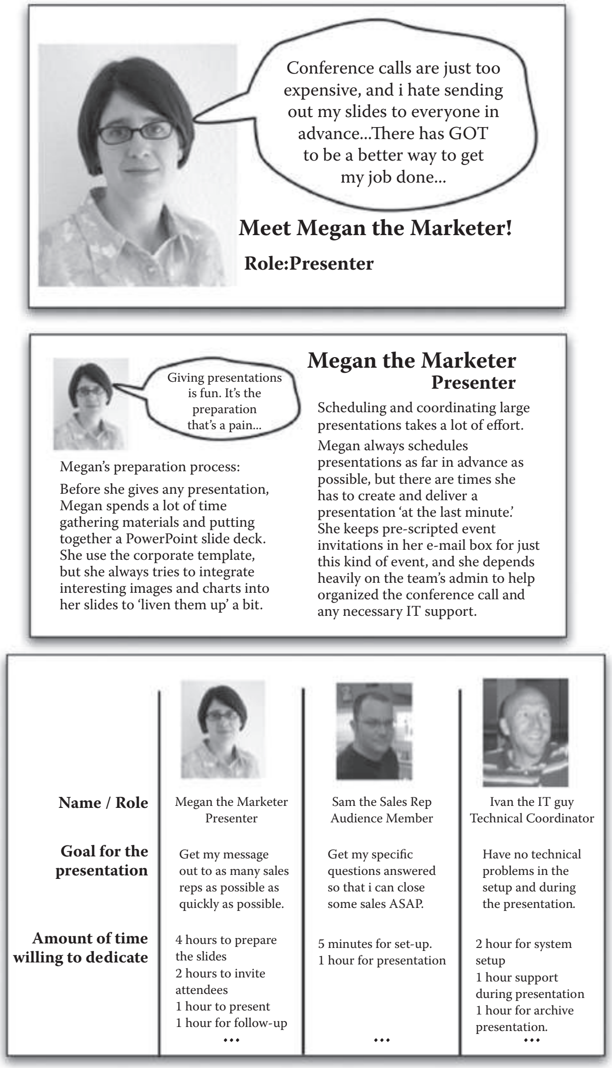
Your persona artifacts should eventually be everywhere around your office (in hallways, coffee room, doors, meeting rooms, stakeholders/leaders' offices, etc.), but they should appear progressively. For every artifact, consider who is going to see it, when (in the development cycle) they are going to see it, and how the environment will affect their ability to digest the information. For example, you might decide to create different "buzz generator" posters (see below for a description) for the developer's hallway versus the marketer's hallway. If you work in place that does not allow posters and such to be displayed around the building, create artifacts that can handed out to individuals, carried around or placed on desk tops.

46.3.3.1.3 Agree on the Information Elements That Should (and Should Not) Be Included on the Artifact

By the time you are ready to create persona artifacts, you will have quite a bit of information about each persona at your disposal. The information you have will all seem highly relevant and deeply interrelated, and it can therefore be difficult to comb out small snippets to include on individual artifacts. Remember that the easiest way to create a useless persona artifact is to overload it with information. For example, you might decide to create "wanted" posters to create buzz and to convey the name, role, and picture for each of your primary personas. It will be tempting to include a quote and maybe a few bulleted details with additional information. Remember your priorities: if you really do want to build buzz and interest, consider limiting the poster to just a photo, a name, and a role. When in doubt, always opt for less information and leave your audience craving more.

46.3.3.1.4 Agree on the Relative Priorities of the Information Elements on the Artifact

Once you decide which information elements should be included on an artifact, prioritize these elements according to how important it is that the element is read and understood. For example, on the "wanted" posters, the photo and name should probably be very large and eye-catching. In contrast, a comparison poster you distribute a few weeks or months later should include names and roles, but these are probably not as important as comparative information about each persona's goals, abilities, desires, and so on. See Figure 46.7, for example, posters in each of the three categories of artifacts.



Conference calls are just too expensive, and i hate sending out my slides to everyone in advance...There has GOT to be a better way to get my job done...

Meet Megan the Marketer!

Role:Presenter

Giving presentations is fun. It's the preparation that's a pain...

Megan the Marketer
Presenter

Scheduling and coordinating large presentations takes a lot of effort.

Megan always schedules presentations as far in advance as possible, but there are times she has to create and deliver a presentation 'at the last minute.' She keeps pre-scripted event invitations in her e-mail box for just this kind of event, and she depends heavily on the team's admin to help organized the conference call and any necessary IT support.

Megan's preparation process:

Before she gives any presentation, Megan spends a lot of time gathering materials and putting together a PowerPoint slide deck. She use the corporate template, but she always tries to integrate interesting images and charts into her slides to 'liven them up' a bit.

Name / Role	Megan the Marketer Presenter	Sam the Sales Rep Audience Member	Ivan the IT guy Technical Coordinator
Goal for the presentation	Get my message out to as many sales reps as possible as quickly as possible.	Get my specific questions answered so that i can close some sales ASAP.	Have no technical problems in the setup and during the presentation.
Amount of time willing to dedicate	4 hours to prepare the slides 2 hours to invite attendees 1 hour to present 1 hour for follow-up ...	5 minutes for set-up. 1 hour for presentation ...	2 hour for system setup 1 hour support during presentation 1 hour for archive presentation. ...

FIGURE 46.7 Examples of a buzz generator, and enricher, and a comparison facilitator poster for the online presentation of software personas.

If you have limited resources (e.g., very little money to use on persona artifacts), think carefully about the artifacts you will need now and try to predict what you will need later. Do not use your entire budget on artifacts you will distribute early; remember that you still face the challenge of keeping the personas alive and useful throughout the Adulthood phase.

46.3.4 Phase 4: Persona Adulthood

Personas are "all grown up" in the adulthood phase and they have a job to do. You have now introduced the personas to the product team and worked to clarify the role and importance of the personas. You have encouraged the product team to embrace the personas and the information they contain, and now it is time to help everyone use the personas to inform the design and development of the product.

Adult personas are ready to be put to work in a variety of ways. Personas can help by answering difficult questions and by focusing activities in a way that takes the guesswork out of making customer-driven decisions. Personas can participate in your product planning, design and development process by

- Being present at your meetings and representing the voice of your customer throughout the development cycle
- Providing consistency by providing a common reference point across your organization, even in a highly chaotic, fast-moving, and ever-changing environment
- Providing a way for all of the product teams to touch base using a common language, and to ensure that everyone is staying focused on creating a good experience on the right audiences

Personas can only be involved and helpful in these ways if they inhabit your workspaces and attend your meetings. Even though they are not real people, personas can become the most powerful voices in the room. Large wall posters, which introduce your personas, should be placed in every meeting room as well as other common spaces where team members discuss product design, features, and overall user experience. The important thing is that the personas' presence is felt. You want your team mates to refer to the personas in their everyday work. Hundreds of tiny decisions are made each day and you want these decisions to be made in consideration of your target audiences.

The more the persona names are used, the more likely it is that everyone in your organization will accept the fact that the personas and the user data they represent are here to stay. The more the persona names replace the word "user" in documents and conversation, the more likely it is that the persona data will shape your product. We suggest that you Ban the word "user" across your organization. Find the owner of each and every document that will help to define or describe your product and encourage them to use the names of the personas instead of the word "user." You have succeeded if you can search the Product Vision Statement, Business Plan, Technical Spec, Marketing and Messaging Plans and find no instance of the word "user."

46.3.4.1 Use Personas for Product Planning, Design, Evaluation, and Release

One obvious way to use personas is to test design ideas by asking questions like: "Will Sally want to use this feature?" We have found that this is only one of the many ways personas can be involved in the product design and development process. The effective persona practitioner must understand the many other ways personas can be involved in existing processes, and ensure that the personas work hard in an organization during the entire development process.

Personas can be used to inform quality assurance test cases, to recruit usability test participants, to make high level and detailed feature priority decisions and to communicate product direction. Personas can also inform marketing, advertising and sales strategy.

46.3.4.1.1 Personas and Product Planning

Personas can help your team envision what your new product should do. Personas can help you understand the context into which you will launch your product and the kinds of problems it needs to solve if it is going to be successful. Now that you have created your personas, you can ask the personas to "tell you their stories"; the needs, goals, and contexts you so carefully included in your persona descriptions will now allow you to generate helpful stories about the way your product will be used and the actions (and reactions) it should elicit.

Personas can help in the product planning process, both by helping you discover important features and by helping you evaluate the relative values of each feature. You can use personas to help you understand and capture your user and system requirements through

- Persona narratives and storytelling
- Persona-focused competitive reviews
- Persona-focused feature brainstorming
- Persona-focused evaluation of proposed features

All of these involve taking on the perspective of your personas to review the competition, your ideas for features, and so on. For example, you can do a persona-focused competitive review by finding out which existing products your new product will compete with. Your marketing team has probably already done this and they are a good source for help with this exercise. If you can, buy a copy of each competing product. (If the products are prohibitively expensive, you can do this exercise using the marketing or collateral materials instead of the actual products; this will give you insight into the reaction of your potential customers to the messages your competitors have deemed to be important.) Once you have access to the products, it is relatively easy to look at them from your personas' perspectives.

Gather the core team and ask one member to "walk through" the competitive products from the perspective of one of you personas. As you observe your "persona" walking through the product, you will find aspects of the product that work well and some that do not work well. If members of your product design team are present, they will come up with ideas for functionality that you must address in your product and ideas for brand new features.

46.3.4.1.2 Personas and Product Design

Once your organization has a vision and overall development plan in place, it is time to design the elements of your product. Your personas helped you understand the big picture, and now they can help you make decisions about specific features and design elements. That is, your personas can show you what these features look and behave like.

There is one process we have found particularly helpful in translating insights derived from personas into ideas for product designs: Design Mapping. We will describe this method in depth in the Section 46.3.4.2 of this chapter so that you can try it out for yourself.

46.3.4.1.3 Personas and Product Design Evaluation

As your team settles in on the features and specific solutions that it needs to embrace to create a successful product, your personas can help in honing the implementation of these features toward the very best design.

You can use your personas to help with evaluation of your features and solutions through

- Cognitive walkthroughs and design reviews with personas
- User testing and ongoing user research with persona profiles
- Quality assurance (QA) testing and bug bashes

The best way to incorporate personas into the above processes is to involve the owner of each process (e.g., the QA manager) to meet with the persona core team. As a rule, meet with process owners earlier than you think is necessary. This will give you time to figure out ways to incorporate personas into the various evaluation processes your company uses before it releases any product.

46.3.4.1.4 Personas and Product Release

Now that your product is getting close to being complete, it is time to turn your attention toward details that are not directly related to product development. You have put a great deal of effort into creating and using your personas to design and build your product. Now that the product is almost complete, your personas (and all of the persona-related materials and tools you have created) can be extremely helpful to those responsible for documenting, supporting, and selling your product.

If you have used the personas throughout the design and development process, you will have many documents that talk about how the product is supposed to work from the point of view of the personas. These documents will be invaluable to the documentation, training, and support professionals in your organization. Marketing and sales professionals will also be able to use the personas (and related deliverables) to help craft materials to support their own work.

Many persona practitioners have told us that the personas seem to "move out" of the design and development offices and "move in" to the documentation, support, and sales offices during product release.

46.3.4.2 A Great Tool for Persona Adulthood: Using Design Maps to Get from Personas to Product Designs

Design Mapping is a process that results in a large flowchart created out of sticky notes, and depicts an individual's end-to-end experience of using a tool or accomplishing a goal. They are artifacts that help you understand and communicate information about the ways people achieve their goals and the ways they could achieve their goals with new tools. Maps are similar to other participatory design tools, but are useful in ways that we have not found other tools to be; Design Maps tell stories about the experiences of personas in the future.

Maps are helpful information gathering and design tools because they are easy to create, iterate, and read. If they are kept in public spaces, they can become an accessible source of insights into user experiences (either as they exist today or as they are envisioned to become). Unlike prose documents or complex flowcharts, Maps make it easy to quickly extract and understand the end-to-end user experience and/or focus on details of interest.

Design Maps can help you test new experiences before you build a new product. Moreover, Design Maps will help you translate what you know about your personas into designs for new experiences that your product could support. Once you design the experience you want to create, it is relatively easy to create features to support that experience.

46.3.4.2.1 What Are Design Maps?

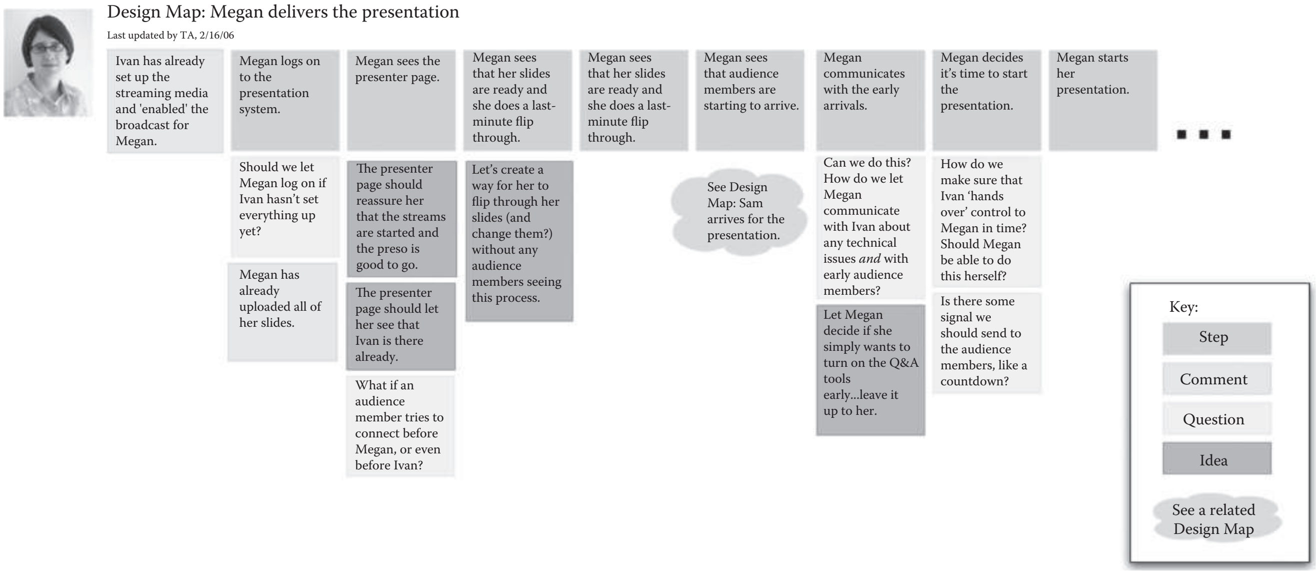
Design Maps tell stories that look into the future; these stories describe how your personas will behave once your new product is built. Those familiar with scenario-based design will recognize that Design Maps have a distinct similarity to scenarios (see Carroll [1995]). Scenarios are short prose stories that describe how aspects of your product will be—or should be—used. Design Maps are both a special type of scenario and a process by which to create scenarios and modify them. Design Maps are flowchart version of many scenarios "strung together" to create a big picture of the experience your product will support. Design Maps are inexpensive (both in terms of time and materials) and they are most helpful when built before paper prototypes and certainly before any code is written.

46.3.4.2.2 Which Processes Should I Design Map?

Design Maps depict end-to-end experiences (not specific features or widgets); you can create Design Maps to explore any experience you want to create as it relates to any aspect of your new product's design. The Design Maps you create should explore the ways that your personas achieve the goals that you have established for them. Remember that their roles and goals may change in your new designs.

46.3.4.2.2.1 Design Map for the "Big Picture" This Map shows the entire experience end-to-end and therefore describes activities in very broad terms. Think of this "overview" Design Map as analogous to a map of the United States with a line drawn on it to show the route of a cross-country driving trip; the overview Map should give the reader a general sense of direction and the order of progression, but should not contain details.

46.3.4.2.2.2 Design Maps for Achieving Major Milestones These Maps should "fit into" the overall Map, but should explore individual goals and tasks more specifically. In the



Design Map: Megan delivers the presentation

Last updated by TA, 2/16/06

Ivan has already set up the streaming media and 'enabled' the broadcast for Megan.

Megan logs on to the presentation system.

Should we let Megan log on if Ivan hasn't set everything up yet?

Megan has already uploaded all of her slides.

Megan sees the presenter page.

The presenter page should reassure her that the streams are started and the preso is good to go.

The presenter page should let her see that Ivan is there already.

What if an audience member tries to connect before Megan, or even before Ivan?

Megan sees that her slides are ready and she does a last-minute flip through.

Let's create a way for her to flip through her slides (and change them?) without any audience members seeing this process.

Megan sees that her slides are ready and she does a last-minute flip through.

Megan sees that audience members are starting to arrive.

See Design Map: Sam arrives for the presentation.

Megan communicates with the early arrivals.

Can we do this? How do we let Megan communicate with Ivan about any technical issues *and* with early audience members?

Let Megan decide if she simply wants to turn on the Q&A tools early...leave it up to her.

Megan decides it's time to start the presentation.

How do we make sure that Ivan 'hands over' control to Megan in time? Should Megan be able to do this herself?

Is there some signal we should send to the audience members, like a countdown?

Megan starts her presentation.

■ ■ ■

FIGURE 46.8 A Design Map exploring how Megan starts an online presentation. Design Maps are created by your team and explore the new experiences you are going to build into your product.

cross-country trip example, a "major milestone" Map would be the equivalent of a highway map showing the roads you used to traverse a single state.

46.3.4.2.2.3 Design Maps for Critical Details These Maps should fit into the milestones Maps much as the milestones fit into the big picture. These Maps explore very specific details of particular tasks, the way an enlargement of a downtown area shows the specific details of the ways streets crisscross a city (Figure 46.8).

46.3.4.2.3 How to Create a Design Map

Maps show steps in a process or experience sequentially, with any questions, comments, or ideas regarding these steps arrayed underneath. Finished Design Maps are large sheets of paper covered with color-coded Post-It™ (or "sticky") notes that describe the user experiences related to your product. Design Maps are created by the design team without the participation of users.

46.3.4.2.3.1 The Elements of a Map: Steps, Questions, Comments, and Design Ideas All Maps have four basic building blocks: steps, questions, comments, and design ideas. Steps should be arrayed horizontally, with related comments, questions, and design ideas arranged under the steps they reference. You can read across the row of steps to get a sense of the process from end to end (i.e., the steps in a task taken to reach a goal), or you can focus on a subset of the steps and read down the columns to understand related questions and ideas (see Figure 46.9).

Steps ("blue" sticky notes): These are the "verbs" or the "backbone" of the process. The facilitator of the Mapping exercise places steps horizontally across the Map. A good way to elicit steps is to ask "What will the persona do next?" Steps are the building blocks of tasks.

Comments ("green" sticky notes): Comments are qualifying statements about steps; they are the most flexible elements on a Map. Comments can describe behaviors, habits, awareness or lack of awareness of features or alternative actions, or even qualities of objects. If you hear an important piece of information, but it is not a step, question or design idea, record it as a green comment.

For example, in our Megan Delivers the Presentation Map (Figure 46.8), the comment "Megan has already uploaded all of her slides" is a note about her actions that could be significant with respect to the rest of the experience. The comment is not a step, but it relates to the step listed above: "Megan logs on to the presentation system." The comment, in this example, serves to remind the facilitator and Map readers that we are assuming that the slides are uploaded in a separate, previous series of actions in this particular Design Map. If this comment were to change (in this example, if we were to change or design the process so that the slides are uploaded immediately before the presentation), it could affect all of the steps, comments, questions, and ideas in the remainder of the Map.

Questions ("yellow" sticky notes): Yellow "questions" are the most useful interview management tool of the Mapper. When you first start Mapping any process, you will identify many questions, some indicating areas where you need clarification and some that express your Mapping participants' issues. In fact, you will probably encounter so many questions that the

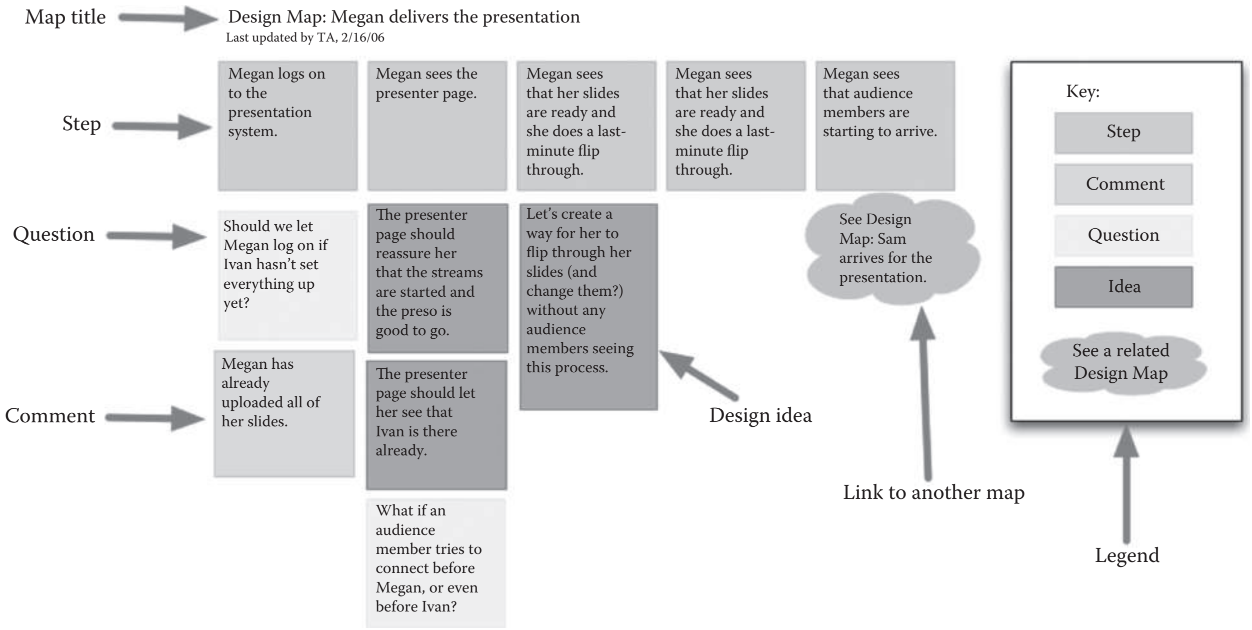

Map title
Design Map: Megan delivers the presentation
Last updated by TA, 2/16/06
Step
Question
Comment
Megan logs on to the presentation system.
Megan sees the presenter page.
Megan sees that her slides are ready and she does a last-minute flip through.
Megan sees that her slides are ready and she does a last-minute flip through.
Megan sees that audience members are starting to arrive.
Should we let Megan log on if Ivan hasn't set everything up yet?
Megan has already uploaded all of her slides.
The presenter page should reassure her that the streams are started and the preso is good to go.
The presenter page should let her see that Ivan is there already.
What if an audience member tries to connect before Megan, or even before Ivan?
Let's create a way for her to flip through her slides (and change them?) without any audience members seeing this process.
See Design Map: Sam arrives for the presentation.
Design idea
Link to another map
Key:
Step
Comment
Question
Idea
See a related Design Map
Legend

FIGURE 46.9 Steps, questions, comments, and design ideas in a design map.

sheer volume and importance of them will threaten to derail your attempt to Map the entire process. Listing the questions on the Map allows you to record and move past them quickly so that you can capture as much of the process as possible without being derailed. Once you create a Map that captures most of an end-to-end process, you can loop back and track down answers to the questions you have identified.

Create a yellow question when

- You have questions about the process
- You are not quite sure what will, should, or could happen next
- Anyone participating in the Mapping session begins to belabor a point

Design Ideas ("pink" sticky notes): As you create your Map, you will inevitably think of, or be presented with, an assortment of ideas for specific features, widgets, or even things like marketing messages. Your Mapping goal is to create a solid picture of the end-to-end experience you want to create for your persona. While you do not want to allow your Mapping session to turn into a discussion of specific new features, you also should not discard good ideas just because they come up at the "wrong time." For example, in Figure 46.8, there is a pink sticky note that says "Let's create a way for her to flip through her slides (and change them?) without any audience members seeing this process." This is an interesting idea and worth capturing. Ask anyone who comes up with a design idea to record it on a pink sticky note, place it on the Map, and move on.

46.3.4.2.3.2 Facilitating a Design Mapping Session Encourage mapping participants to focus on the *experience*, not on the *tool*. The goal is not to have a Map that tells you "the serial number registration tracking database will feed the score records to the page via ASP," but one that says "Megan can see that Ivan is already logged on."

During Design Mapping sessions, remind your team to consider:

- Do the tasks assigned to personas in the Design Map correspond to your personas' skills? For example, if Sam the Sales Rep is expected to spend an hour doing technical preparation so that he can attend the presentation, he is not being well served by the new design. If Megan's Design Map allows her to answer questions by presenting new content on the fly, you are on the right track.
- Does this new process being constructed in the Map offer undeniable advantages to the personas over the old way of doing things?
- Are we assuming things have to be done a certain way just because that is the way they are done now?

As you move through a Mapping session, remember to table questions that might sidetrack your work by providing everyone with yellow sticky notes and encouraging participants to write down difficult questions and issues and post them on the map. During your Design Mapping session, you might hear a comment like: "Well, if we are assuming Sam can access the Internet using a broadband connection from wherever he is, even if he is on the road, we can assume he will not have any trouble viewing the presentation. It would also be great if we could assume he has all the media players installed, and the latest version of the browsers. This makes things easier for us." This is a good opportunity to refer to your personas. Do you have information about Sam's technical setup? How likely is it that his computer will have the latest media players? Is this likely to be something that is done for him, or something he has to figure out for himself? Your personas will be able to immediately answer some of these questions; others will have to go onto to the Map to be answered later. In this particular example, you might decide to create another Map that assumes that Sam does not have everything he needs loaded on his machine. What process are you going to design to make it easy for him to prepare for and attend a presentation in this case?

Sometimes you will want to move fairly quickly, placing blue steps across the top of the Map, and filling in details later. Other times your team might find it most effective to hash out the details under each step, before moving to the next one. In either case, you will want to limit mapping sessions to 2–3 hours each.

After each session, follow-up on any questions or issues raised and add answers to the Map. It can be useful to convert the sticky notes paper versions of your Maps into electronic versions in Microsoft Visio or a similar tool. This makes it easy for participants to review progress and quickly scan for new material. The electronic versions are useful for printing in various formats and sending to stakeholders for review at a distance.

46.3.4.2.4 Use Design Maps to Create Wireframes

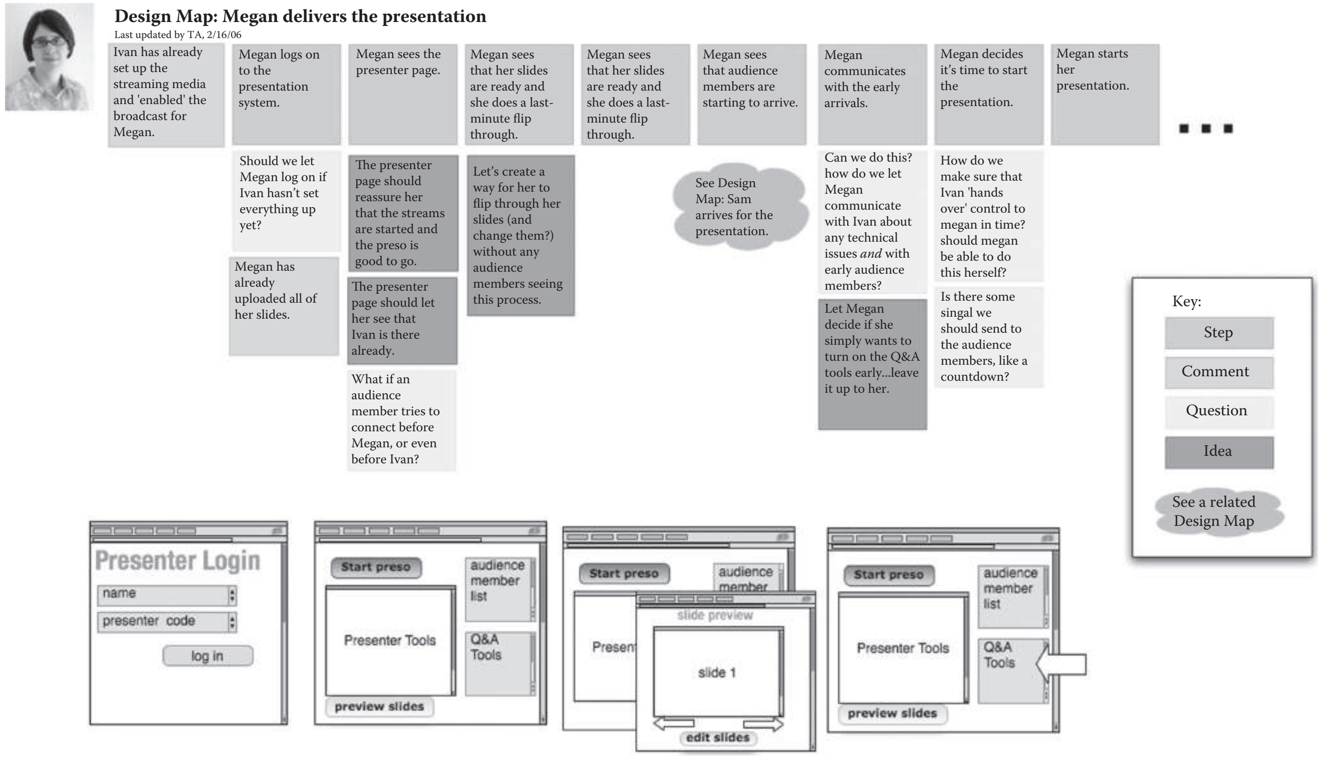
Wireframes tend to evolve naturally from Design Maps. Once you and your team have agreed on the experience you want to facilitate for your personas, it is relatively easy to use the steps, assumptions, questions, and design ideas in the Maps to create wireframes of the product's UI.

With your team, identify the columns of the Design Map that "go together" and should be grouped on a single interface (see Figure 46.10). At this point, the UI designer or graphic designer should be heavily involved. Consider what information you are collecting from the personas and when you are collecting it so that you can plan to display it on the UI at the right times (e.g., if you have not asked the persona for their name yet, you cannot create a wireframe for a personalized interface).

46.3.4.2.5 The Benefits of Design Maps

Perhaps the most important benefit of Design Maps (and wireframes) is that they hold and communicate a shared vision of the project for your entire team. Seeing the "big picture" early on in the project helps motivate everyone toward the common goal of making it real. Design Maps enable your entire team to "see" the product from the personas' point of view, giving the architects of the product the opportunity to understand and empathize with users. A deeper understanding of the planned product and personas early on in the development process can enhance each team member's work on your product—whether they are coding, marketing, managing, testing, funding or selling it.

Design Maps (and their associated wireframes) are a perfect document to work from when communicating project plans to stakeholders. Once your Design Maps are completed, you can use them to perform design walk-throughs of your product with other team members. Have someone read through the map and another person check the prototype or



Design Map: Megan delivers the presentation

Last updated by TA, 2/16/06

Ivan has already set up the streaming media and 'enabled' the broadcast for Megan.

Megan logs on to the presentation system.

Should we let Megan log on if Ivan hasn't set everything up yet?

Megan has already uploaded all of her slides.

Megan sees the presenter page.

The presenter page should reassure her that the streams are started and the preso is good to go.

The presenter page should let her see that Ivan is there already.

What if an audience member tries to connect before Megan, or even before Ivan?

Megan sees that her slides are ready and she does a last-minute flip through.

Let's create a way for her to flip through her slides (and change them?) without any audience members seeing this process.

Megan sees that her slides are ready and she does a last-minute flip through.

Megan sees that audience members are starting to arrive.

See Design Map: Sam arrives for the presentation.

Megan communicates with the early arrivals.

Can we do this? how do we let Megan communicate with Ivan about any technical issues *and* with early audience members?

Let Megan decide if she simply wants to turn on the Q&A tools early...leave it up to her.

Megan decides it's time to start the presentation.

How do we make sure that Ivan 'hands over' control to megan in time? should megan be able to do this herself?

Is there some singal we should send to the audience members, like a countdown?

Megan starts her presentation.

■ ■ ■

Key:

Step

Comment

Question

Idea

See a related Design Map

FIGURE 46.10 Wireframes evolve naturally from design maps. You and your mapping team can evaluate which process steps belong together and design the interface flow accordingly.

the product to make sure the Mapped process is supported by the products' design.

As your development team begins coding your new product, check in frequently to evaluate what they are building against what you *thought* they were going to build. Use the Design Maps to "test" the emerging product: Does the product being built actually support the experience you have designed? The Design Maps you created define a task flow; when the product changes during development, use the Maps to evaluate whether or not the changes support the tasks you identified.

SIDEBAR 46.6 Design Mapping Illuminates—and Fixes!—Holes in Other Kinds of Specification Documents

Raina Brody and Sylvia Olveda

Usability Specialists, Amazon.com

The Amazon usability team works with many different project teams from all over the company. For one of our projects, we were working with a team that was designing and developing a new feature for the website. The team had already created three primary personas and a very large document full of use cases, which specified many features and UI elements. The use cases and UI specifications all referred to their personas by name. In short, it looked like they had done a lot of good, user-centered work and were pretty far along the path toward creating a good solution. We asked the designer and the design manager why they felt they needed our help.

They replied "all of this stuff looks really good, but there are enormous holes in the use cases when it comes to the actual UI, the experience flow, and the technical implementation/specifications. Some of the technical services and features these use cases call for do not exist and simply cannot be built—and the rest of the team does not seem to get this." We decided to lead a few Design Mapping Sessions to help them identify and document the holes and to communicate their concerns to the rest of the team.

We set up a meeting with the project team (including members of the business team, design and development teams) and asked them to pick a couple of the most important use cases. They were a bit hesitant at first, because some members felt that they had already done all of this work in the use case document. After describing their primary use cases, we started the Design Mapping exercise. Very soon, it became clear that there were holes in the existing documentation. For example, some of the original use cases assumed the existence of data types and search services that simply did not exist, and that were not scoped as part of the project schedule. The Design Map helped us discover that very important elements in their use cases had not yet been thought about.

The Design Mapping process really started going well when the project team members realized that we, the usability team, were not trying to design their product for them. Instead, we were serving as facilitators to help them come up with and discuss their ideas from a customer experience and technical feasibility perspective. Of course, it was our job to ask (and sometimes answer) some difficult questions about whether customers would be able to discover and/or understand some of the UI elements the team came up with.

The Mapping process generated excitement from the project team and some of the best ideas came from developers. After our first meetings, the Mapping sessions went well and went very fast. The team agreed that the Design Mapping process helped them identify what was hard, what was easy, and what they could really do. The team also realized that while the original use case document contained a lot of good ideas, the complexities of the actual implementation were not made as apparent as they were with the Mapping session.

At the end of the project, we asked the team how the Design Mapping and personas helped with the project. The answers we got were very encouraging:

From a key developer: "Design mapping helped everyone on the team to really focus on the project from our customer's point of view. It took a lot of disparate opinions and ideas about what we wanted to build and provided a common focus. It was responsible for some very productive brainstorming."

From the Technical Program Manager: "The personas and Mapping process are invaluable tools designing a product that is customer-centric; together they have the power to overcome designs driven by emotion and bias: two things commonly found in every product design cycle."

From the Product Manager: "The Design Mapping we did at the beginning of our project has proven to be an invaluable tool to keep the team focused on what is best for our target customers. It was also an extremely helpful team-building exercise."

From the team Manager: "The Design Mapping process helps us develop the right features for the right customers. Project teams are hesitant at first because it seems like a significant time investment, but once they go through the process, they find it invaluable. It actually saves time in the long run by making sure teams make better decisions up front."

WE LEARNED SOME VALUABLE LESSONS FROM THIS MAPPING EXERCISE

If you are Mapping with a team that has never tried it before, do not start by Mapping a controversial feature or experience. Instead, start with one of the features or experiences that seems well defined and clear. This will help them team understand the Mapping process and will show them that you do not intend to take over the design of their product.

Encourage teams to do use cases after they do Design Maps. Design Maps generate lots of use cases, and they have the added benefit of clearly illustrating how use cases fit together in an end to end customer or user experience.

Present Design Mapping as a technique to facilitate the product team's design process. It is important that the team still feel in control, even in a situation where you are leading a Mapping exercise.

Create a Mapping plan after your first Mapping meeting. After you do one Map, it will be easy to identify other user experiences you want to Map. Revisit your Mapping schedule after you do the first two or three Maps to integrate new ideas for Maps and a new perspective on how long the Mapping process will take.

Think about who you want to invite and who you need to invite to Mapping sessions. We always facilitated, and invited at least one designer, the technical program or project manager, between one and three members of the technical/development team, and the product manager.

Word is spreading fast, and now we have so many requests to facilitate Design Mapping sessions we can hardly keep up with them.

46.3.5 Phase 5: Persona Lifetime Achievement and Retirement

Once the project or product is completed, it is time to think about what has been accomplished and prepare for the next project. You will want to assess how effective the persona method was for your team and product development process. If you are beginning to think about the next product (or next version of the product you have just released) you will need to decide whether, and how, you will reuse your existing personas and the information they contain.

The end of a product cycle is a good time to assess the effectiveness of personas for the team and to take stock of lessons learned for the next time. How did the development team accept the method? Were your personas useful? To what extent were they accurate and precise? We will provide suggestions and tools you can use to validate the use of personas in the development process and to determine if the persona effort was worth the effort and resources it required. Did personas change the product? Did they change your design and development process? User-centered designers are constantly under pressure to validate the worth and ROI of their activities and personas can be useful tools for measuring the success of both the product and of the UCD activities as a whole.

Recall the questions we recommended you answer in Phase 1, Family Planning, related to "organizational introspection." Assessing ROI is, in simple terms, working to answer those original questions (listed in the left column of Figure 46.11) in terms that are meaningful now that you are done with the effort (i.e., those questions listed in the right column of Figure 46.11).

46.3.5.1 Retirement

Depending on the nature of your products, you might be able to reuse the personas or "reincarnate" some of your persona data in new personas. That is, you will need to decide what to do with your "old" personas as you prepare for your next project. Do your personas retire, do they change over time? Do they buy your product and start using it? Can other product teams utilize your personas or some portion of the information in them (i.e., are they reincarnated)?

In most cases, you will decide to use some combination of direct reuse (using them again without alteration), reincarnation (reusing some their content and related data) or retirement (discarding or completely replacing some of the personas). Note that, if your persona effort has been a success, retirement and reincarnation can be a bit tricky; to have "room" for the next set of personas, you will need to help your organization let go of the personas they have come to know so well.

Questions You Should Ask During Family Planning	Questions You Will Ask During Lifetime Achievement
What resources do we have for personas and other UCD activities?	How much did the persona effort actually cost?
What product problems do we want to solve with personas?	Has the product improved? How much, and in what ways?
What process problems do we want to solve with personas?	Has the process improved? In what ways?
How can we ensure that the personas will be accepted and used by our colleagues?	Were personas perceived as helpful? Has the company's focus on users improved? In what ways?

FIGURE 46.11 The work you did during family planning is highly related to what you will need to do in lifetime achievement.

Before you decide what to do with your personas, you need to revisit the data sources you used to create them. If you are about to start work on the next version of the product you have just released, it is likely that many (but probably not all) of your data sources are still relevant and you can reuse entire personas or some of the information in the personas. If you are moving on to create a completely different product or if there have been major shifts in strategy, then perhaps only a few of the data sources (e.g., those that relate to your company or to the general product space in which you work) may still be relevant.

46.3.5.2 Reusing Your Personas

If you are building a new version of your product, or a new product for the same audience, you might find that many of your personas can be reused. Your personas could be reused by the same team that used them originally, by a new design and development team, or perhaps by a team in some other part of your company, like marketing, sales, or product support.

When you created your personas, you assimilated your data, created persona skeletons, prioritized the skeletons, and built some or all of the skeletons into full personas. When you move on to the next version of your product, you can reevaluate the primary versus secondary classification for each of your original personas. You might decide to demote one of your primary personas and promote one of your secondary personas; this promotion/demotion is especially useful if you are building a new version of the same project but your company has decided to focus on a slightly different user base. Additionally, you can revisit some of the persona skeletons you created but never developed for the first project; it is possible that one of these sketches would be just right for the new project. If so, you have a tremendous head start and can simply build up the sketch into a full persona.

46.3.5.3 Reincarnating Personas

If some of the data in your personas is still relevant, but the personas you originally created are not, you can create "reincarnated" personas by reusing "old" data in new personas. If the products you develop serve users in a specific market segment or industry, you will find many data sources that stay relevant no matter what project you are working on. If the data sources are still relevant, but the particular personas you have created are not, it is important to do some research and find some additional sources.

Once you have collected the appropriate set of data (which will include sources you have already used), you can revisit the processes described in the Conception & Gestation section and reassimilate data points according to the issues you are finding related to your new project.

46.3.5.4 Retiring Your Personas

You might decide that you do not want to reuse or reincarnate your personas or their underlying data at all. There are many reasons to retire personas before moving on to the next project, such as the following:

- The current project is significantly different than the last project.
- The users' goals have changed.
- Your company adopted a new strategy or are targeting a different user base.
- There are significant changes to the environment in which your product will be used, such as new technologies or new competitive products that have "changed the landscape?" (e.g., the advent of streaming media and broadband in the home, or Bluetooth technology for computing).

When you determine that a persona or set of personas is no longer relevant, it is a good idea to *officially* retire the persona(s) before moving on.

Why "officially?" Why not just take down the posters and start working on a new set of personas? Because if you have done your job well, you have made the personas incredibly memorable and all your work has paid off; people in your organization have absorbed various amounts of information about the primary personas and are accustomed to thinking about them. If you try to introduce a new set of personas on top of an old set, you run the risk of confusing your team—which will destroy the clarity that personas are supposed to provide.

You cannot reach into your colleagues' heads and erase everything they know about the personas you have been using, but you have to find a way to help them move past the old personas and let go of the (no longer relevant) information they contain. This can be as simple as an e-mail announcement that the old set of personas is retiring and why. If you are moving on to a totally different product or a new strategy, most of your colleagues would know about the switch and the retirement announcement will make sense to them. Use this as an opportunity to invite feedback from the team on the ways personas helped or did not help them do their jobs; you can use this feedback to tweak your customized persona lifecycle the next time around.

46.3.5.5 Moving on to the Next Persona Effort

Finally, it is worth noting that the activities we recommend at the end of one persona project function to prepare you for the next persona project; regaining control over the personas and evaluating the success of the effort will help you be even more successful in your next effort. The lifetime achievement, reuse, and retirement phase provides an excellent opportunity to touch base with your core team members and with other stakeholders to talk about how things went. As you dive back into Family Planning you will want to predict the new issues you will encounter. This final lifecycle phase is a great time to have a postmortem to talk about what improved, stayed the same, or worsened during or due to your persona effort.

46.4 SUMMARY

Personas can be an invaluable asset to product design and development teams, but they must be created and used with care. The best advice we can give you as you embark on your own persona efforts is to keep in mind the following axioms that are at the heart of the persona lifecycle approach:

- Building personas from assumptions is good; building personas from data is much, much better.
- Personas are a highly memorable, inherently usable communication tool *if* they are communicated well.
- Personas can be initiated by executives or first used as part of a bottom-up, grass roots experiment but eventually need support at all levels of an organization.
- Personas are not a stand-alone UCD process but should be integrated into existing processes and used to augment existing tools.
- Personas can help bring a user-centered focus into even a stubbornly technology-focused organization.
- Effective persona efforts requires organizational introspection and strategic thinking.
- Personas *can* be created fast and show their value quickly, but if you want to get the full value of personas, you will have to commit to a significant investment of time and resources.

In short, the "devil is in the details" when it comes to launching a persona effort within an organization. Perhaps this is the reason that persona efforts are inevitably more work than you think they are going to be. If you can dedicate some time, and can get some help from at least one colleague, the persona lifecycle will enable you to integrate persona creation and use methods into your existing processes. Equally importantly, it will provide you with data-driven, user-focused reasons to change aspects of existing processes that are not working well for the team and are resulting in problematic products.

SIDEBAR 46.7 Personas in the Present—and Looking toward the Future

Kim Goodwin

Vice President and General Manager, Cooper

As for the future of personas, I have heard a number of people say that personas are just the latest fad, and no one will care about them in a few years. It is true that some people are using things they call "personas" just because they are the latest craze, and no doubt the popularity of personas will fade quickly with those people who find that a photo, a name, and a collection of personal tidbits do not accomplish much. People who look to personas to solve all of their problems will also be disappointed. However, those practitioners who take the time to master personas—which are easy to understand but not so easy to develop and use—will find that personas are one tool they will never put down.

In the last few years, Cooper designers have been using personas for more than product definition and interaction design guidance. We find that experience goals (how a persona wants to feel, as opposed to the end goals we usually focus on with personas) are invaluable in guiding the aesthetic aspects of the visual, auditory, and physical design. Does your online banking persona want to feel her money is secure? A lime green background and Comic Sans typeface probably are not the best choices. Does a critically ill person want to be reminded of his condition? No, so perhaps his home healthcare device should not look as if it belongs in a hospital. We have used them for designing business processes, organizations, and training curricula, as well.

Personas have been adopted by certain parts of companies, but not yet by others. In many cases, we see companies that use personas for one product but not others, or who rely heavily on personas to design their websites, but do not yet use them to inform the design of other customer-facing systems. I believe the lack of adoption is not due to a failing of the method, but to a common failing of organizations: silos. The website or product design teams are generally isolated from other parts of a company, so those other groups are unaware of the advances those teams are making. In addition, there is seldom anyone with overall responsibility for a customer's or user's experience with that company. To realize the true potential of their investment in personas, companies will need to look for other opportunities to use them, even outside of electronic systems. For example, a couple of years ago, I spoke with a human resources publication interested in applying employee personas to benefits plans.

There is no question in my mind: personas are here to stay, and we will see them showing up in more and more places. After all, personas are about understanding our fellow human beings, and that will never go out of style.

46.4.1 Areas for Further Investigation

While we are clearly fans of the persona approach, we know that personas are not foolproof. In fact, our own experiences with them have proven to us that personas have their limits—personas are not a panacea and they are not right for every situation or organization. The details of these shortcomings are not well understood at this time, though as noted earlier, some discussion, case studies and research does exist to lead the way (e.g., Blomquist and Arvola 2002; Chapman and Milham 2006; Chapman et al. 2008; Rönkkö et al. 2004). Clearly, personas do not help a development team build better products merely by their presence. As we argue here (and in our books, *The Persona Lifecycle*, 2006; *The Essential Persona Lifecycle*, 2010), personas must be assimilated by the product team and explicitly used in design and development activities. But, what uses of personas have the most value? What characteristics of personas are the most critical? We have provided some process and guidance to help practitioners employ the persona approach successfully, but at present, this is based more on practice and experience than science. In addition to the research and case studies cited previously, we need more rigorous evaluations of the method to better understand its benefits and limitations. Some reasonable suggested directions for research are provided by Chapman and Milham (2006) and Miaskiewicz and Kozar (2006).

The process we recommend for persona creation, while accomplishable and generally effective, is not particularly precise. Given the same data as input, different groups of people will likely create different personas. Many practitioners (e.g., Miaskiewicz, Sumner, and Kozar 2008; McGinn and Kotamraju 2008; Mulder and Yaar 2007; Sinha 2003; including us) have proposed quantitative analysis methods as the basis for persona creation to enhance repeatability and accuracy. And while this is certainly a reasonable move in the right direction, it is the thoughtful application of qualitative data that helps make a persona into a persona. Thus, we do not believe a quantitative approach could ever appropriately solve the problem alone. For similar reasons, the notion that personas can be abstracted into reusable elemental characteristics or directly repurposed across multiple product lines seems imprudent. Again, more investigation along these lines is needed.

Finally, there are some cases where development teams have attempted to create representations of things other than people (e.g., organizational archetypes—a persona of a company) to be used, like personas, in the development of products or services. We believe that such representations, while potentially useful, do not have the power and impact that representations of people have. They are not as memorable or provocative as personas—we relate to personas in ways that we cannot relate to other abstractions. Thus, we suggest this as another interesting line of research.

REFERENCES

Adlin, T., and J. Pruitt. 2010. *The Essential Persona Lifecycle: Your Guide to Building and Using Personas*. Burlington, MA: Morgan Kaufmann Press.

Antle, A. N. 2006. Child-personas: Fact or fiction? In *Proceedings of DIS06: Designing Interactive Systems: Processes, Practices, Methods, & Techniques 2006*, 22–30. New York: ACM.

Blomquist, Å., and M. Arvola. 2002. "Personas in Action: Ethnography in an interaction design team." In *Proceedings of the Second Nordic Conference on Human-Computer Interaction, NordiCHI*. Aarhus, Denmark. New York: ACM Press.

Carroll, J., ed. 1995. *Scenario-Based Design: Envisioning Work and Technology in System Development*. New York: Wiley.

Carroll, J. 2000a. Five reasons for scenario-based design. *Interact Comput* 13:43–60.

Carroll, J. 2000b. *Making use: Scenario-Based Design of Human-Computer Interactions*. Cambridge, MA: MIT Press.

Chang, Y., Y. Lim, and E. Stolterman. 2008. Personas: From theory to practices. In *Proceedings of NordiCHI 2008: Using Bridges*, 439–442. Lund, Sweden. New York: ACM.

Chapman, C. N., and R. P. Milham. 2006. The personas' new clothes: Methodological and practical arguments against a popular method. In *Proceedings of the Human Factors and Ergonomics Society (HFES) 50th Annual Conference*, 634–636. San Francisco, CA: Human Factors and Ergonomics Society.

Chapman, C. N., E. Love, R. P. Milham, P. ElRif, and J. L. Alford. 2008. Quantitative evaluation of personas as information. In *Proceedings of the Human Factors and Ergonomics Society 52nd Annual Meeting*, 1107–11, New York. San Francisco, CA: Human Factors and Ergonomics Society.

Constantine, L., and L. Lockwood. 2001. Personas. For Use Newsletter, August 15. http://www.foruse.com/newsletter/foruse15.htm.

Constantine, L., and L. Lockwood. 2002. Modeling: Persona Popularity and Role Relationships. For Use Newsletter, October 26. http://www.foruse.com/newsletter/foruse26.htm.

Cooper, A. 1999. *The Inmates are Running the Asylum*. Indianapolis, IN: Macmillan.

Cooper, A., and R. Reimann. 2003. *About Face 2.0: The Essentials of Interaction Design*. Wiley Publishing, Inc.

Dantin, U. 2005. Application of personas in user interface design for educational software. In *Proceedings of Australasian Computing Education Conference-ACE*, 239–47. Darlinghurst, Australia: Australian Computer Society, Inc.

Dharwada, P. 2006. *Use of Personas for User Interface Design: Results of Field and Experimental Studies*. PhD Thesis, Clemson University.

Dharwada, P., J. S. Greenstein, A. K. Gramopadhye, and S. J. Davis. 2007. A case study on use of personas in design and development of an audit management system. In *Human Factors and Ergonomics Society Annual Meeting Proceedings (September 2007)*, 469–73.

Dreyfuss, H. 1955. *Designing for People*. Republished 2003, New York: Allworth Press.

Freed, J. 2004. *Ahead of the Game: Best Buy Revamps its Stores to be Ready for the Challenge from a New Line of Electronics Retailers*. Associated Press. http://www.projo.com/business/content/projo_20040520_best20x.201cc9.html.

Grudin, J. 1990. Constraints in product development organizations. In *Proceedings of Participatory Design Conference*, 14–21.

Grudin, J. 1993. Obstacles to participatory design in large product development organizations. In *Participatory Design: Principles and Practices*, ed. D. Schuler and A. Namioka, 99–119. Mahwah, NJ: Erlbaum.

Grudin, J., and J. Pruitt. 2002. Personas, participatory design, and product development: An infrastructure for engagement. In *Proceedings of PDC 2002*, 144–61. Boston, MA: PDC.

Haikara, J. 2007. Usability in agile software development: Extending the interaction design process with personas approach. In *Extreme Programming-XP Universe, LNCS 4536*, ed. G. Concas et al., 153–6. Berlin: Springer-Verlag.

Hill, V., and V. Bartek. 2007. Telling the user's story. In *Proceedings of CHIMIT'07*, Cambridge, MA. New York: ACM.

Hourihan, M. 2002. Taking the "you" out of user: My experience using Personas. Boxes and arrows. http://boxesandarrows.com/archives/002330.php.

Jacobson, I. 1995. "The use-case construct in object-oriented software engineering". In *Scenario-Based Design*, ed. J. M. Carroll. New York: Wiley.

Jacobson, I., M. Christerson, P. Jonsson, and G. Övergaard. 1992. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Reading, MA: Addison-Wesley.

Junior, P., and L. Filgueiras. 2005. User modeling with personas. In *CLIHC '05: Proceedings of the 2005 Latin American Conference on Human-Computer Interaction (2005)*, 277–82. New York: ACM

Khalayli, N., S. Nyhus, K. Hamnes, and T. Terum. 2007. Persona based rapid usability kick-off. In *Proceedings of CHI 2007*, San Jose, California. New York: ACM.

Kujala, S., and M. Kauppinen. 2004. Identifying and selecting users for user-centered design. In *Proceedings of NordiCHI'04*, Tampere, Finland. New York: ACM.

Levinson, M. 2003. Website redesign: How to Play to Your Audience. CIO.com http://www.cio.com/archive/111503/play.html.

Long, F. 2009. Real or imaginary: The effectiveness of using personas in product design. In *Irish Ergonomics Review, Proceedings of the IES Conference 2009*, 1–10. Dublin.

Markensten, E., and H. Artman. 2004. Procuring usable systems using unemployed personas. In *Proceedings of NordiCHI'04*, Tampere, Finland. New York: ACM.

McGinn, J., and N. Kotamraju. 2008. Data-driven persona development. In *Proceedings of CHI 2008*, Florence, Italy. New York: ACM.

McQuaid, H. L., A. Goel, and M. McManus. 2003. When you can't talk to customers: Using storyboards and narratives to elicit empathy for users. In *Proceedings of the 2003 International Conference on Designing Pleasurable Products and Interfaces, 2003*, Pittsburgh, PA. New York: ACM.

Mello, S. 2003. *Customer-Centric Product Definition: The Key to Great Product Development*. Boston, MA: PDC Professional Publishing.

Miaskiewicz, T., and K. Kozar. 2006. The use of the delphi method to determine the benefits of the personas method – an approach to systems design. In *Proceedings of SIGHCI 2006*. Paper 7.

Miaskiewicz, T., T. Sumner, and K. Kozar. 2008. A latent semantic analysis methodology for the identification and creation of personas. In *Proceedings of CHI 2008*, Florence, Italy. New York: ACM.

Moore, G. A. 1991. *Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers*. New York: Harper Collins Publishers. (Revised in 2002).

Mulder, S., and Z. Yaar. 2007. *The User Is Always Right: A Practical Guide to Creating and Using Personas for the Web*. Berkeley, CA: New Riders.

Nielsen, J. 1993. *Usability Engineering*. Boston, MA: Academic Press.

Nieters, J. E., S. Ivaturi, and I. Ahmed. 2007. Making personas memorable. In *Proceedings of CHI 2007*, San Jose, California. New York: ACM.

Panke, S., B. Gaiser, and B. Werner. 2007. Evaluation as Impetus for Innovations in E-learning—Applying Personas to the Design of Community Functions. *MERLOT J Online Learn Teach* (3) 2.

Pruitt, J., and T. Adlin. 2006. *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*. Burlington, MA: Morgan Kaufmann Press.

Rönkkö, K., M. Hellman, B. Kilander, and Y. Dittrich. 2004. "Personas is not applicable: Local remedies interpreted in a wider context." In *Proceedings of the Participatory Design Conference 2004*, 27–31 Toronto, Canada. New York: ACM.

Siegel, D. A. 2010. The mystique of numbers: Belief in quantitative approaches to segmentation and persona development. In *Proceedings of CHI 2010*, Atlanta, GA. New York: ACM.

Sinha, R. 2003. Persona development for information-rich domains. In *Proceedings of CHI 2003*. New York: ACM Press.

Sissors, J. 1966. What is a market. *J Mark* 30:17–21.

Shyba, L., and J. Tam. 2005. Developing character personas and scenarios: Vital steps in theatrical performance and HCI goal-directed design. In *Proceedings of C&C'05*, London, United Kingdom. New York: ACM.

Star, S. L., and J. R. Griesemer. 1989. Institutional ecology, 'translations,' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907–1939. *Social Studies of Science* 19:387–420.

Triantafyllakos, G., G. Palaigeorgiou, and I. A. Tsoukalas. 2009. Design alter egos: Constructing and employing fictional characters in collaborative design sessions. In *Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology*. Swinton, UK: British Computer Society.

Tychsen, A., and A. Canossa. 2008. Defining personas in games using metrics. In *Proceedings of Future Play 2008*, 73–80, Toronto, Ontario, Canada. New York: ACM.

Upshaw, L. 1995. *Building Brand Identity: A Strategy for Success in a Hostile Marketplace*. New York: John Wiley & Sons.

Weinstein, A. 1998. *Defining your Market: Winning Strategies for High-Tech, Industrial, and Service Firms*. New York: Haworth Press.

47 Prototyping Tools and Techniques

Michel Beaudouin-Lafon and Wendy E. Mackay

CONTENTS

47.1 INTRODUCTION

“A good design is better than you think.”

Heftman
as cited in Raskin 2000

Design is about making choices. In many fields that require creativity and engineering skill, such as architecture and automobile design, prototypes both inform the design process and help designers select the best solution.

This chapter describes tools and techniques for using prototypes to design interactive systems. The goal is to illustrate how they can help designers generate and share new ideas, get feedback from users or customers, choose among design alternatives, and articulate reasons for their final choices.

We begin with our definition of a prototype and then discuss prototypes as design artifacts, introducing four dimensions for analyzing them. We then discuss the role of prototyping within the design process, in particular the concept of a design space and how it is expanded and contracted by generating and selecting design ideas. The following sections describe specific prototyping approaches: rapid prototyping, both offline and online, for early stages of design, and iterative and evolutionary prototyping, which use online development tools. Finally, we address the specific issue of prototyping mixed with reality and pervasive computing systems.

47.1.1 What Is a Prototype?

We define a *prototype* as a concrete representation of part or all of an interactive system. A prototype is a tangible artifact, not an abstract description that requires interpretation. Designers, as well as managers, developers, customers, and end users, can use these artifacts to envision and to reflect upon the final system.

Note that other fields may define prototype differently. For example, an architectural prototype is a scaled-down model of the final building. This is not possible for prototypes of interactive systems: The designer may limit the amount of information the prototype can handle, but the actual user interface must be presented at full scale. Thus, a prototype interface to a database may handle only a small subset of the final database but must still present a full-size display and interaction techniques. Fashion designers create another type of prototype, a full-scale, one-of-a-kind model, such as a handmade dress sample. Although in haute couture this prototype may also be the final product, the ready-to-wear market requires additional design phases to create a design that can be mass-produced in a range of sizes. Some interactive system prototypes begin as one-of-a-kind models that are then distributed widely (since the cost of duplicating software is so low). However, most successful software prototypes evolve into the final product and then continue to evolve as new versions of the software are released.

Hardware and software engineers often create prototypes to study the feasibility of a technical process. They conduct systematic, scientific evaluations with respect to predefined benchmarks and, by systematically varying parameters, fine tune the system. Designers in creative fields, such as typography or graphic design, create prototypes to express ideas and reflect on them. This approach is intuitive, oriented more to discovery and generation of new ideas than to evaluation of existing ideas.

HCI is a multidisciplinary field that combines elements of science, engineering, and design (Mackay and Fayard 1997; Dijkstra-Erikson, Mackay, and Arnowitz 2001). Prototyping is primarily a design activity, although we use software engineering to ensure that software prototypes evolve into technically sound working systems and we use scientific methods to study the effectiveness of particular designs.

47.2 PROTOTYPES AS DESIGN ARTIFACTS

We can look at prototypes as both concrete artifacts in their own right or as important components of the design process. When viewed as artifacts, successful prototypes have several characteristics: They support creativity, helping the developer to capture and generate ideas, facilitate the exploration of a design space, and uncover relevant information about users and their work practices. They encourage communication, helping designers, engineers, managers, software developers, customers, and users to discuss options and interact with each other. They also permit early evaluation since they can be tested in various ways, including traditional usability studies and informal user feedback, throughout the design process.

We can analyze prototypes and prototyping techniques along the following four dimensions:

- *Representation* describes the form of the prototype, such as sets of paper sketches or computer simulations.
- *Precision* describes the level of detail at which the prototype is to be evaluated, such as informal and rough or highly polished.
- *Interactivity* describes the extent to which the user can actually interact with the prototype, such as “watch only” or fully interactive.
- *Evolution* describes the expected lifecycle of the prototype, such as throwaway or iterative.

47.2.1 Representation

Prototypes serve different purposes and thus take different forms. A series of quick sketches on paper can be considered a prototype; so can a detailed computer simulation. Both are useful; both help the designer in different ways. We distinguish between two basic forms of representation: offline and online.

Offline prototypes (also called “paper prototypes”) do not require a computer. They include paper sketches, illustrated storyboards, cardboard mock-ups, and videos. The most salient characteristics of offline prototypes (of interactive systems) is that they are created quickly, usually in the early stages of design, and they are usually thrown away when they have served their purposes.

Online prototypes (also called “software prototypes”) run on a computer. They include computer animations, interactive video presentations, programs written with scripting languages, and applications developed with interface builders. The cost of producing online prototypes is usually higher, and may require skilled programmers to implement advanced interaction and visualization techniques or to meet tight performance constraints. Software prototypes are usually more effective in the later stages of design, when the basic design strategy has been decided.

In our experience, programmers often argue in favor of software prototypes even at the earliest stages of design. Because they are already familiar with a programming language, these programmers believe it will be faster and more useful to write code than to “waste time” creating paper prototypes. In 20 years of prototyping, in both research and industrial settings, we have yet to find a situation in which this is true.

First, offline prototypes are very inexpensive and quick. These permit a very rapid iteration cycle and help prevent the designer from becoming overly attached to the first possible solution. Offline prototypes make it easier to explore the design space, examining a variety of design alternatives and choosing the most effective solution. Online prototypes introduce an intermediary between the idea and the implementation, slowing down the design cycle.

Second, offline prototypes are less likely to constrain how the designer thinks. Every programming language or development environment imposes constraints on the interface, limiting creativity and restricting the number of ideas considered. If a particular tool makes it easy to create scrollbars and pull-down menus and difficult to create a zoomable interface, the designer is likely to limit the interface accordingly. Considering a wider range of alternatives, even if the developer ends up using a standard set of interface widgets, usually results in a more effective design.

Finally, and perhaps most importantly, offline prototypes can be created by a wide range of people, not just programmers. Thus all types of designers, technical or otherwise, as well as users, managers, and other interested parties, can all contribute on an equal basis. Unlike programming software, modifying a storyboard or cardboard mockup requires no particular technical skill. Collaborating on paper prototypes not only increases participation in the design process, but also improves communication among team members and increases the likelihood that the final design solution will be well accepted.

Although we believe strongly in offline prototypes, they are not a panacea. In some situations, they are insufficient to evaluate fully a particular design idea. For example, interfaces requiring rapid feedback to users or complex, dynamic visualizations usually require software prototypes. However, particularly when using video and Wizard of Oz techniques, offline prototypes can be used to create very sophisticated representations of the system.

Prototyping is an iterative process and all prototypes provide information about some aspects while ignoring others. The designer must consider the purpose of the prototype (Houde and Hill 1997) at each stage of the design process and choose the representation that is best suited to the current design question.

47.2.2 Precision

Prototypes are explicit representations that help designers, engineers and users reason about the system being built. By their nature, prototypes require details. A verbal description such as "the user opens the file" or "the system displays the results" provides no information about what the user actually does or sees. Prototypes force designers to show the interaction: just how does the user open the file and what are the specific results that appear on the screen?

Precision refers to the relevance of details with respect to the purpose of the prototype.* For example, when sketching a dialog box, the designer specifies its size, the positions of each field and the titles of each label. However not all these details are relevant to the goal of the prototype: it may be necessary to show where the labels are, but too early to choose the text. The designer can convey this by writing nonsense words or drawing squiggles, which shows the need for labels without specifying their actual content.

Although it may seem contradictory, a detailed representation need not be precise. This is an important characteristic of prototypes: those parts of the prototype that are not precise are those open for future discussion or for exploration of the design space. Yet they need to be incarnated in some form so the prototype can be evaluated and iterated.

The level of precision usually increases as successive prototypes are developed and more and more details are set. The forms of the prototypes reflect their level of precision: sketches tend not to be precise, whereas computer simulations are usually very precise. Graphic designers often prefer using hand sketches for early prototypes because the drawing style can directly reflect what is precise and what is not: the wiggly shape of an object or a squiggle that represents a label are directly perceived as imprecise. This is more difficult to achieve with an online drawing tool or a user interface builder.

The form of the prototype must be adapted to the desired level of precision. Precision defines the tension between what the prototype states (relevant details) and what the prototype leaves open (irrelevant details). What the prototype states is subject to evaluation; what the prototype leaves open is subject to more discussion and design space exploration.

47.2.3 Interactivity

An important characteristic of HCI systems is that they are interactive: users both respond to them and act upon them. Unfortunately, designing effective interaction is difficult: many interactive systems (including many websites) have a good look but a poor feel. HCI designers can draw from a long tradition in visual design for the former, but have relatively little experience with how interactive software systems should be used: personal computers have only been commonplace for a couple decades. Another problem is that the quality of interaction is tightly linked to the end users and a deep understanding of their work practices: a word processor designed for a professional typographer requires a different interaction design than one designed for secretaries, even though ostensibly they serve similar purposes. Designers must take the context of use into account when designing the details of the interaction (Beaudouin-Lafon 2004).

A critical role for an interactive system prototype is to illustrate how the user will interact with the system. While this may seem more natural with online prototypes, in fact it is often easier to explore different interaction strategies with offline prototypes. Note that interactivity and precision are orthogonal dimensions. One can create an imprecise prototype that is highly interactive, such as a series of paper screen images in which one person acts as the user and the other

* Note that the terms low-fidelity and high-fidelity prototypes are often used in the literature. We prefer the term precision because it refers to the content of the prototype itself, not its relationship to the final, as-yet-undefined system.

plays the system. One may create a very precise but noninteractive prototype, such as a detailed animation that shows feedback from a specific action by a user.

Prototypes can support interaction in various ways. For off-line prototypes, one person (often with help from others) plays the role of the interactive system, presenting information and responding to the actions of another person playing the role of the user. For online prototypes, parts of the software are implemented, while others are played by a person. (This approach, called the "Wizard of Oz" after the character in the 1939 movie of the same name, is explained in a later section.) The key is that the prototype feels interactive to the user.

Prototypes can support different levels of interaction. Fixed prototypes, such as video clips or precomputed animations, are noninteractive: the user cannot interact, or pretend to interact, with it. Fixed prototypes are often used to illustrate or test scenarios. Fixed-path prototypes support limited interaction. The extreme case is a fixed prototype in which each step is triggered by a prespecified user action. For example, the person controlling the prototype might present the user with a screen containing a menu. When the user points to the desired item, he or she presents the corresponding screen showing a dialogue box. When the user points to the word "OK," he or she presents the screen that shows the effect of the command. Even though the position of the click is irrelevant (it is used as a trigger), the person playing the role of the user gets the feel of the interaction. Of course, this type of prototype can be much more sophisticated, with multiple options at each step. Fixed-path prototypes are very effective with scenarios and can be used for horizontal and task-based prototypes (see section on prototyping strategies below).

Open prototypes support large sets of interactions. Such prototypes work like the real system, with some limitations. They usually cover only part of the system (see vertical prototypes) and often have limited error handling or reduced performance relative to that of the final system.

Prototypes may thus illustrate or test different levels of interactivity. Fixed prototypes simply illustrate what the interaction might look like. Fixed-path prototypes provide designers and users with the experience of what the interaction might feel like, but only in prespecified situations. Open prototypes allow designers and users to explore a wide range of possible forms of interaction with the system.

47.2.4 Evolution

Prototypes have different life spans: Rapid prototypes are created for a specific purpose and then thrown away, iterative prototypes evolve, either to work out some details (increasing their precision) or to explore various alternatives, and evolutionary prototypes are designed to become part of the final system.

Rapid prototypes are especially important in the early stages of design. They must be inexpensive and easy to produce, since the goal is to quickly explore a wide variety of possible types of interaction and then throw them away. Note that rapid prototypes may be offline or online. Creating precise software prototypes, even if they must be reimplemented in the final version of the system, is important for detecting and fixing interaction problems. A later section presents specific prototyping techniques, both offline and online.

Iterative prototypes are developed as a reflection of a design in progress, with the explicit goal of evolving through several design iterations. Designing prototypes that support evolution is sometimes difficult. There is a tension between evolving toward the final solution and exploring an unexpected design direction, which may be adopted or thrown away completely. Each iteration should inform some aspect of the design. Some iterations explore different variations of the same theme. Others may systematically increase precision, working out the finer details of the interaction. A later section describes tools and techniques for creating iterative prototypes.

Evolutionary prototypes are a special case of iterative prototypes in which the prototype evolves into part or all of the final system (Figure 47.1). Obviously, this only applies to software prototypes. Extreme Programming (Beck 2000) advocates this approach, tightly coupling design and implementation

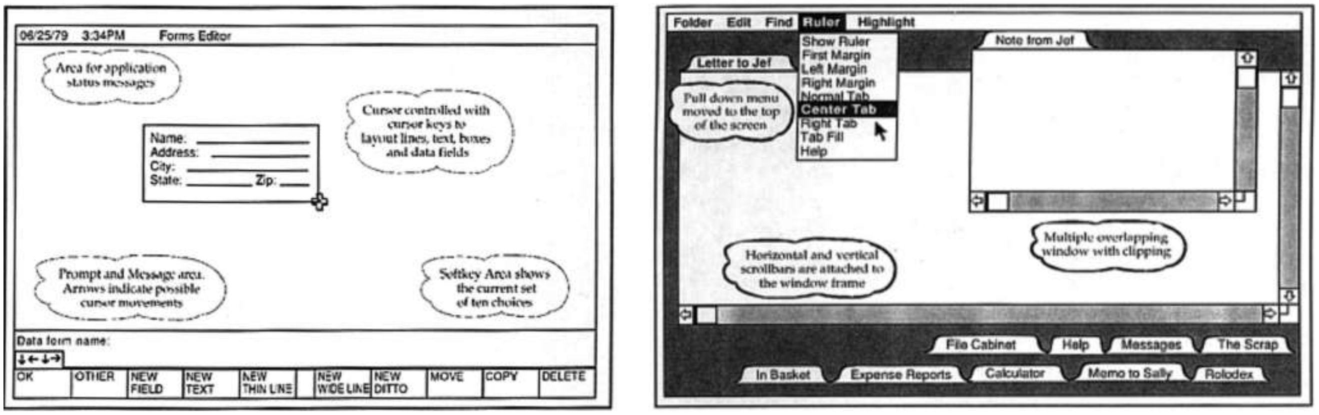

06/25/79 3:34PM Forms Editor
Area for application status messages
Cursor controlled with cursor keys to layout lines, text, boxes and data fields
Name:
Address:
City:
State:
Zip:
Prompt and Message area. Arrows indicate possible cursor movements
Softkey Area shows the current set of ten choices
Data form name:
↓←↓→
OK
OTHER
NEW FIELD
NEW TEXT
NEW THIN LINE
NEW WIDE LINE
NEW DITTO
MOVE
COPY
DELETE
Folder Edit Find Ruler Highlight
Show Ruler
First Margin
Left Margin
Right Margin
Normal Tab
Center Tab
Right Tab
Tab Fill
Help
Letter to Jef
Note from Jef
Pull down menu moved to the top of the screen
Multiple overlapping window with clipping
Horizontal and vertical scrollbars are attached to the window frame
File Cabinet
Help
Messages
The Scrap
In Basket
Expense Reports
Calculator
Memo to Sally
Rolodex

FIGURE 47.1 Evolutionary prototypes of the Apple Lisa: July 1979 (left), October 1980 (right). (From Perkins, R., D. S. Keller, and F. Ludolph. 1997. Inventing the Lisa user interface. *ACM Interact* 4(1):40–53. With permission.)

and building the system through constant evolution of its components. Evolutionary prototypes require more planning and practice than the approaches above because the prototypes are both representations of the final system and the final system itself, making it more difficult to explore alternative designs. We advocate a combined approach, beginning with rapid prototypes and then using iterative or evolutionary prototypes according to the needs of the project. A later section describes how to create iterative and evolutionary prototypes, using appropriate development tools.

47.3 PROTOTYPES AND THE DESIGN PROCESS

In the previous section, we looked at prototypes as artifacts, that is, the results of a design process. Prototypes can also be seen as artifacts for design, as an integral part of the design process. Prototyping helps designers think: prototypes are the tools they use to solve design problems. In this section, we focus on prototyping as a process and its relationship to the overall design process.

47.3.1 User-Centered Design

The field of HCI is both user centered (Norman and Draper 1986) and iterative. User-centered design places the user at the center of the design process, from the initial analysis of user requirements to testing and evaluation. Prototypes support this goal by allowing users to see and experience the final system long before it is built. Designers can identify functional requirements, usability problems, and performance issues early and improve the design accordingly.

Iterative design involves multiple design-implement-test loops,* enabling the designer to generate different ideas and successively improve upon them. Prototypes support this goal by allowing designers to evaluate concrete representations of design ideas and select the best.

Prototypes reveal the strengths as well as the weaknesses of a design. Unlike pure ideas, abstract models, or other representations, they can be contextualized to help understand how the real system would be used in a real setting. Because prototypes are concrete and detailed, designers can explore different real-world scenarios and users can evaluate them with respect to their current needs. Prototypes can be compared directly with other, existing systems, and designers can learn about the context of use and the work practices of the end users. Prototypes can help designers reanalyze the user's needs during the design process: not abstractly, as with traditional requirements analysis, but in the context of the system being built.

47.3.2 Participatory Design

Participatory design is a form of user-centered design that actively involves the user in all phases of the design process (see Greenbaum and Kyng 1991). Users are not simply consulted at the beginning and called in to evaluate the system at the end; they are treated as partners throughout. This early and active involvement of users helps designers avoid unpromising design paths and develop a deeper understanding of the actual design problem. Obtaining user feedback at each phase of the process also changes the nature of the final evaluation, which is used to fine tune the interface rather than discover major usability problems.

A common misconception about participatory design is that designers are expected to abdicate their responsibilities as designers, leaving the design to the end user. In fact, the goal is for designers and users to work together, each contributing their strengths to clarify the design problem as well as explore design solutions. Designers must understand what users can and cannot contribute. Usually, users are best at understanding the context in which the system will be used and subtle aspects of the problems that must be solved. Innovative ideas can come from both users and designers, but the designer is responsible for considering a wide range of options that might not be known to the user and balancing the trade-offs among them.

Because prototypes are shared, concrete artifacts, they serve as an effective medium for communication within the design team as well as with users. We have found that collaborating on prototype design is an effective way to involve users in participatory design. Prototypes help users articulate their needs and reflect on the efficacy of design solutions proposed by designers.

47.3.3 Exploring the Design Space

Design is not a natural science: the goal is not to describe and understand existing phenomena but to create something new. Of course, designers benefit from scientific research findings, and they may use scientific methods to evaluate interactive systems. However, designers also require specific techniques for generating new ideas and balancing complex sets of trade-offs to help them develop and refine design ideas.

Designers from fields such as architecture and graphic design have developed the concept of a design space, which constrains design possibilities along some dimensions, while leaving others open for creative exploration. Ideas for the design space come from many sources: existing systems, other designs, other designers, external inspiration, and accidents that prompt new ideas. Designers are responsible for creating a design space specific to a particular design problem. They explore this design space, expanding and contracting it as they add and eliminate ideas. The process is iterative: more cyclic than reductionist. That is, the designer does not begin with a rough idea and successively add more details that are precise until the final solution is reached. Instead, he or she begins with a design problem, which imposes a set of constraints, and then generates a set of ideas to form the initial design space. He or she then explores this design space, preferably with the user, and selects a particular design direction to pursue. This closes

* Software engineers refer to this as the Spiral model (Boehm 1988).

off part of the design space, but opens up new dimensions that can be explored. The designer generates additional ideas along these dimensions, explores the expanded design space, and then makes new design choices. Design principles (e.g., Beaudouin-Lafon and Mackay 2000) help this process by guiding it both in the exploration and choice phases. The process continues, in a cyclic expansion and contraction of the design space, until a satisfying solution is reached or time has run out.

All designers work with constraints: not just limited budgets and programming resources, but also design constraints. These are not necessarily bad: one cannot be creative along all dimensions at once. However, some constraints are unnecessary, derived from poor framing of the original design problem. If we consider a design space as a set of ideas and a set of constraints, the designer has two options. She can modify ideas within the specified constraints or modify the constraints to enable new sets of ideas. Unlike traditional engineering, which treats the design problem as a given, designers are encouraged to challenge, and if necessary, change the initial design problem. If she reaches an impasse, the designer can either generate new ideas or redefine the problem (and thus change the constraints). Some of the most effective design solutions derive from a more careful understanding and reframing of the design brief.

Note that all members of the design team, including users, may contribute ideas to the design space and help select design directions from within it. However, it is essential that these two activities are kept separate. Expanding the design space requires creativity and openness to new ideas. During this phase, everyone should avoid criticizing ideas and concentrate on generating as many as possible. Clever ideas, half-finished ideas, silly ideas, and impractical ideas all contribute to the richness of the design space and improve the quality of the final solution. In contrast, contracting the design space requires critical evaluation of ideas. During this phase, everyone should consider the constraints and weigh the trade-offs. Each major design decision must eliminate part of the design space: rejecting ideas is necessary in order to experiment and refine others and make progress in the design process. Choosing a particular design direction should spark new sets of ideas, and those new ideas are likely to pose new design problems. In summary, exploring a design space is the process of moving back and forth between creativity and choice.

Prototypes aid designers in both aspects of working with a design space: generating concrete representations of new ideas and clarifying specific design directions. The next two sections describe techniques that have proven most useful in our own prototyping work, both for research and product development.

47.3.4 Expanding the Design Space: Generating Ideas

The most well-known idea generation technique is brainstorming, introduced by Osborn (1957). His goal was to create synergy within the members of a group: ideas suggested by one participant would spark ideas in other participants. Subsequent studies (Collaros and Anderson 1969; Diehl and Stroebe 1987) challenged the effectiveness of group brainstorming, finding that aggregates of individuals could produce the same number of ideas as groups. They found certain effects, such as production blocking, free riding, and evaluation apprehension, were sufficient to outweigh the benefits of synergy in brainstorming groups. Since then, many researchers have explored different strategies for addressing these limitations. For our purpose, the quantity of ideas is not the only important measure: the relationships among members of the group are also important. As de Vreede et al. (2000) pointed out, one should also consider elaboration of ideas, as group members react to each other's ideas.

We have found that brainstorming, including a number of variants, is an important group-building exercise for participatory design. Of course, designers may brainstorm ideas by themselves. However, brainstorming in a group is more enjoyable and, if it is a recurring part of the design process, plays an important role in helping group members share and develop ideas together.

The simplest form of brainstorming involves a small group of people. The goal is to generate as many ideas as possible on a prespecified topic: quantity, not quality, is important. Brainstorming sessions have two phases: the first for generating ideas and the second for reflecting upon them. The initial phase should last no more than an hour. One person should moderate the session, keeping time, ensuring that everyone participates, and preventing people from critiquing each other's ideas. Discussion should be limited to clarifying the meaning of a particular idea. A second person records every idea, usually on a flipchart or transparency on an overhead projector. After a short break, participants are asked to reread all the ideas and each person marks their three favorite ideas.

One variation is designed to ensure that everyone contributes, not just those who are verbally dominant. Participants write their ideas on individual cards or Post-it notes for a prespecified period. The moderator then reads each idea aloud. Authors are encouraged to elaborate (but not justify) their ideas, which are then posted on a whiteboard or flipchart. Group members may continue to generate new ideas, inspired by the others they hear.

We use a variant of brainstorming, called "video brainstorming" (Mackay 2000), as a very fast technique for prototyping interaction: instead of simply writing or drawing their ideas, participants act them out in front of a video camera (Figure 47.2). The goal is the same as other brainstorming exercises, that is, to create as many new ideas as possible, without critiquing them. However, the use of video, combined with paper or cardboard mock-ups, encourages participants to experience the details of the interaction and to understand each idea from the perspective of the user, while preserving a tangible record of the idea.

Each video brainstorming idea should take two to five minutes to generate and capture, allowing participants to simulate a wide variety of ideas very quickly. The resulting video clips

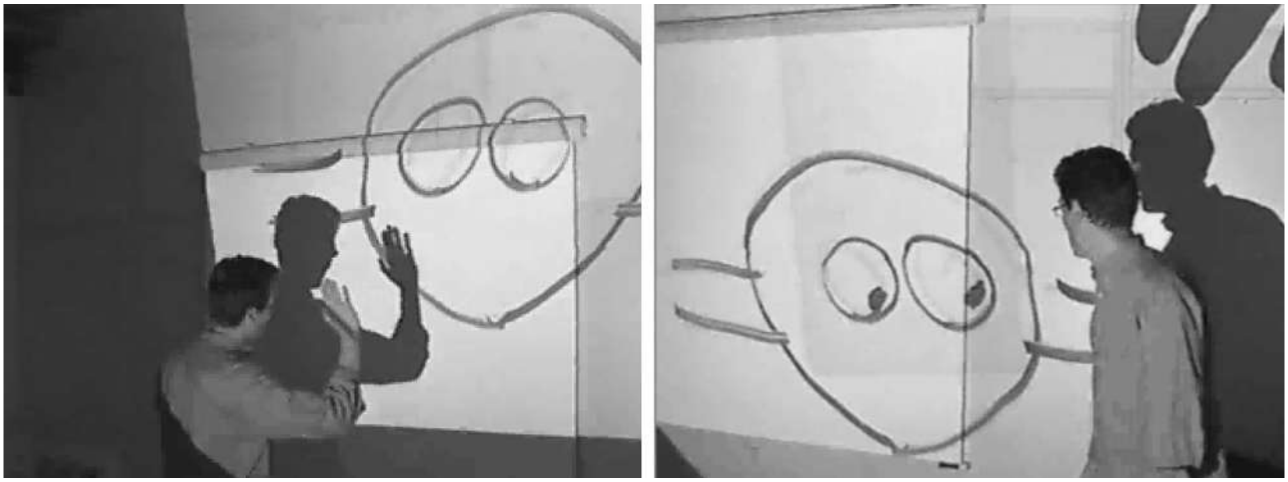


FIGURE 47.2 Video brainstorming an animated character. One participant uses an overhead projector to project an image on the wall and responds to the actions of a second participant, who plays the role of the user. Here, the animated character, a very rough sketch on a transparency, reponds when the user waves and moves its eyes to follow the user.

provide illustrations of each idea that are easier to understand (and remember) than hand-written notes. (We find that raw notes from brainstorming sessions are not very useful after a few weeks because the participants no longer remember the context in which the ideas were created, whereas video brainstorming clips are useful years later.)

Video brainstorming requires thinking more deeply about each idea than in traditional oral brainstorming. It is possible to stay vague and general when describing an interaction in words or even with a sketch, but acting out the interaction in front of the camera forces the author of the idea (and the other participants) to consider seriously the details of how a real user would actually interact with the idea. Video brainstorming also encourages designers and users to think about new ideas in the context in which they will be used. We also find that video clips from a video brainstorming session, even though rough, are much easier for the design team to interpret than written ideas from a standard brainstorming session.

We generally run a standard brainstorming session, either orally or with cards, prior to a video brainstorming session, in order to maximize the number of ideas to be explored. Participants then take their favorite ideas from the previous session and develop them further as video brainstorms. Each person is asked to direct at least two ideas, incorporating the hands or voices of other members of the group. We find that, unlike standard brainstorming, video brainstorming encourages even the quietest team members to participate.

47.3.5 Contracting the Design Space: Selecting Alternatives

After expanding the design space by creating new ideas, designers must stop and reflect upon the choices available to them. After exploring the design space, designers must evaluate their options and make concrete design decisions: choosing some ideas, specifically rejecting others, and leaving other aspects of the design open to further idea generation activities. Rejecting good, potentially effective ideas is difficult, but necessary to make progress.

Prototypes often make it easier to evaluate design ideas from the user's perspective. They provide concrete representations that can be compared. Many of the evaluation techniques described elsewhere in this book could be applied to prototypes, to help focus the design space. The simplest situation is when the designer must choose among several discrete, independent options. Running a simple experiment, using techniques borrowed from psychology, allows the designer to compare how users respond to each of the alternatives. The designer builds a prototype, with either fully implemented or simulated versions of each option. The next step is to construct tasks or activities that are typical of how the system would be used, and ask people from the user population to try each of the options under controlled conditions. It is important to keep everything the same, except for the options being tested.

Designers should base their evaluations on both quantitative measures, such as speed or error rate, and qualitative measures, such as the user's subjective impressions of each option. Ideally, of course, one design alternative will be clearly faster, prone to fewer errors, and preferred by the majority of users. More often, the results are ambiguous, and the designer must consider other factors when making the design choice. (Interestingly, running small experiments often highlights other design problems and may help the designer reformulate the design problem or change the design space.)

The more difficult (and common) situation is when the designer faces a complex, interacting set of design alternatives, in which each design decision affects a number of others. Designers can use heuristic evaluation techniques, which rely on our understanding of human cognition, memory, and sensory-perception. They can also evaluate their designs with respect to ergonomic criteria or design principles (Beaudouin-Lafon and Mackay 2000).

Another strategy is to create one or more scenarios that illustrate how the combined set of features will be used in a realistic setting. The scenario must identify who is involved, where the activities take place, and what the user does over a specified period. Good scenarios involve more than a string of independent tasks; they should incorporate real-world activities, including common or repeated tasks, successful activities, breakdowns, and errors, with both typical and unusual events. The designer then creates a prototype that simulates or implements the aspects of the system necessary to illustrate each set of design alternatives. Such prototypes can be tested by asking users to walk through the same scenario several times, once for each design alternative. As with experiments and usability studies, designers can record both quantitative and qualitative data, depending on the level of the prototypes being tested.

The previous section described an idea-generation technique called "video brainstorming," which allows designers to generate a variety of ideas about how to interact with the future system. We call the corresponding technique for focusing in on a design "video prototyping." Video prototyping can incorporate any of the rapid-prototyping techniques (offline or online) described in a later section. Video prototypes are quick to build, force designers to consider the details of how users will react to the design in the context in which it will be used, and provide an inexpensive method of comparing complex sets of design decisions.

To an outside observer, video brainstorming and video prototyping techniques look very similar: both involve small design groups working together, creating rapid prototypes, and interacting with them in front of a video camera. Both result in video illustrations that make abstract ideas concrete and help team members communicate with each other. The critical difference is that video brainstorming expands the design space, by creating a number of unconnected collections of individual ideas, whereas video prototyping contracts the design space, by showing how a specific collection of design choices work together in a single design proposal.

47.3.6 Prototyping Strategies

Designers must decide what role prototypes should play with respect to the final system and in which order to create different aspects of the prototype. The next section presents four strategies: horizontal, vertical, task oriented, and scenario based, which focus on different design concerns. These strategies can use any of the prototyping techniques covered in the following sections.

47.3.6.1 Horizontal Prototypes

The purpose of a horizontal prototype is to develop one entire layer of the design at the same time. This type of prototyping is most common with large software development teams, where designers with different skill sets address different layers of the software architecture. Horizontal prototypes of the user interface are useful to get an overall picture of the system from the user's perspective and address issues such as consistency (similar functions are accessible through similar user commands), coverage (all required functions are supported), and redundancy (the same function is/is not accessible through different user commands).

User interface horizontal prototypes can begin with rapid prototypes and progress through to working code. Software prototypes can be built with an interface builder without creating any of the underlying functionality, making it possible to test how the user will interact with the user interface without worrying about how the rest of the architecture works. However, some level of scaffolding or simulation of the rest of the application is often necessary; otherwise, the prototype cannot be evaluated properly. Consequently, software horizontal prototypes tend to be evolutionary, that is, they are progressively transformed into the final system.

47.3.6.2 Vertical Prototypes

The purpose of a vertical prototype is to ensure that the designer can implement the full, working system, from the user interface layer down to the underlying system layer. Vertical prototypes are often built to assess the feasibility of a feature described in a horizontal, task-oriented, or scenario-based prototype. For example, when we developed the notion of magnetic guidelines in the CPN2000 system to facilitate the alignment of graphical objects (Beaudouin-Lafon and Mackay 2000, Beaudouin-Lafon and Lassen 2000), we implemented a vertical prototype to test not only the interaction technique but also the layout algorithm and the performance. We knew that we could only include the particular interaction technique if we could implement a sufficiently fast response.

Vertical prototypes are generally high precision software prototypes because their goals are to validate ideas at the system level. They are often thrown away because they are generally created early in the project, before the overall architecture has been decided, and they focus on only one design question. For example, a vertical prototype of a spelling checker for a text editor does not require text-editing functions to be implemented and tested. However, the final version will need to be integrated into the rest of the system, which may involve considerable architectural or interface changes.

47.3.6.3 Task-Oriented Prototypes

Many user interface designers begin with a task analysis to identify the individual tasks that the user must accomplish with the system. Each task requires a corresponding set of functionality from the system. Task-based prototypes are organized as a series of tasks, which allows both designers and users to test each task independently, systematically working through the entire system.

Task-oriented prototypes include only the functions necessary to implement the specified set of tasks. They combine the breadth of horizontal prototypes, to cover the functions required by those tasks, with the depth of vertical prototypes, enabling detailed analysis of how the tasks can be supported. Depending on the goal of the prototype, both offline and online representations can be used for task-oriented prototypes.

47.3.6.4 Scenario-Based Prototypes

Scenario-based prototypes are similar to task-oriented ones, except that they do not stress individual, independent tasks, but rather follow a more realistic scenario of how the system would be used in a real-world setting. Scenarios are stories that describe a sequence of events and how the user reacts. A good scenario includes both common and unusual situations, and should explore patterns of activity over time. Bødker (1999) developed a checklist to ensure that no important issues have been left out.

We find it useful to begin with anecdotes derived from observations of our interviews with real users. Ideally, some of those users should participate in the creation of specific use and other users should critique them based on how realistic they are. Use scenarios are then turned into design scenarios, in which the same situations are described but with the functionality of the proposed system. Design scenarios are used, among other things, to create scenario-based video prototypes or software prototypes. Like task-based prototypes, the developer needs to write only the software necessary to illustrate the components of the design scenario. The goal is to create a situation in which the user can experience what the system would be like in a realistic use context, even if it addresses only a subset of the planned functionality.

The following section describes a variety of rapid prototyping techniques that can be used in any of these four prototyping strategies. We begin with offline rapid prototyping techniques, followed by online prototyping techniques.

47.4 RAPID PROTOTYPING

The goal of rapid prototyping is to develop prototypes very quickly, in a fraction of the time it would take to develop a working system. By shortening the prototype-evaluation cycle, the design team can evaluate more alternatives and iterate the design several times, improving the likelihood of finding a solution that successfully meets the user's needs. Rapid prototypes also serve to cut off unpromising design directions, saving time, and money. It is far easier to reject an idea based on a rapid prototype than a more fully developed one, and one reduces the chance of spending a great deal of time and effort on a design that ultimately does not work.

How rapid is rapid depends on the context of the particular project and the stage in the design process. Early prototypes, such as sketches, can be created in a few minutes. Later in the design cycle, a prototype produced in less than a week may still be considered rapid if the final system is expected to take months or years to build. Precision, interactivity, and evolution all affect the time it takes to create a prototype. Not surprisingly, a precise and interactive prototype takes more time to build than an imprecise or fixed one.

The techniques presented in this section are organized from most rapid to least rapid, according to the representation dimension previously introduced. Offline techniques are generally more rapid than online techniques. However, creating successive iterations of an online prototype may end up being faster than creating new offline prototypes.

47.4.1 Offline Rapid Prototyping Techniques

Offline prototyping techniques range from simple to very elaborate. Because they do not involve software, they are usually considered a tool for thinking through the design issues, to be thrown away when they are no longer needed. This section describes simple paper and pencil sketches, three-dimensional mock-ups, Wizard of Oz simulations, and video prototypes.

47.4.1.1 Paper and Pencil

The fastest form of prototyping involves paper, transparencies and Post-it notes to represent aspects of an interactive system (e.g., Muller 1991). By playing the roles of both the user and the system, designers can get a quick idea of a wide variety of different layout and interaction alternatives, in a very short period of time (Rettig 1994; Snyder 2003).

Designers can create a variety of low-cost special effects. For example, a tiny triangle drawn at the end of a long strip cut from an overhead transparency makes a handy mouse pointer, which can be moved by a colleague in response to the user's actions. Post-it notes, with prepared lists, can provide pop-up menus. An overhead projector pointed at a whiteboard makes it easy to project transparencies (hand drawn or preprinted, overlaid on each other as necessary) to create an interactive display on the wall. The user can interact by pointing (Figure 47.3) or drawing on the whiteboard. One or more people can watch the user and move the transparencies in response to her actions. Everyone in the room gets an immediate impression of how the eventual interface might look and feel.

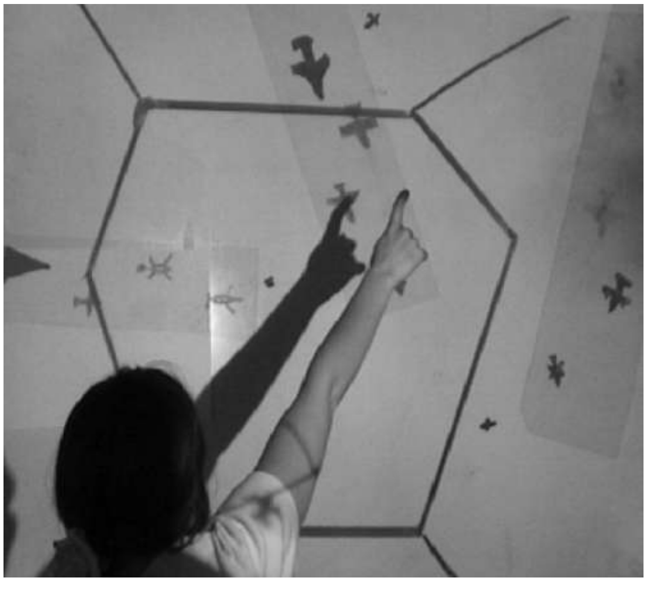


FIGURE 47.3 Hand-drawn transparencies can be projected onto a wall, creating an interface a user can respond to.

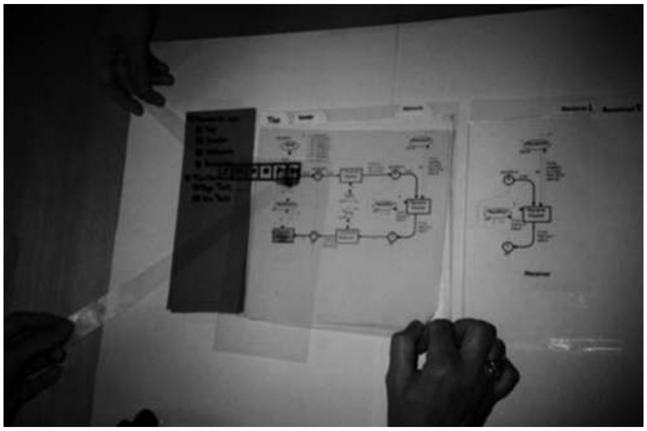


FIGURE 47.4 Several people work together to simulate interacting with this paper prototype. One person moves a transparency with a mouse pointer while another moves the diagram accordingly.

Note that most paper prototypes begin with quick sketches on paper, then progress to more carefully drawn screen images made with a computer (Figure 47.4). In the early stages, the goal is to generate a wide range of ideas and expand the design space, not to determine the final solution. Paper and pencil prototypes are an excellent starting point for horizontal, task-based, and scenario-based prototyping strategies.

47.4.1.2 Mock-Ups

Architects use mock-ups or scaled prototypes to provide three-dimensional illustrations of future buildings. Mock-ups are also useful for interactive system designers, helping them move beyond two-dimensional images drawn on paper or transparencies (see Bødker et al. 1988). Generally made of cardboard, foamcore, or other found materials (Frishberg 2006), mock-ups are physical prototypes of the new system. Figure 47.5 shows an example of a handheld mock-up showing the interface to a new handheld device. The mock-up provides a deeper understanding of how the interaction will work in real-world situations than a set of screen images.

Mock-ups allow the designer to concentrate on the physical design of the device, such as the position of buttons or the screen. The designer can also create several mock-ups and compare input or output options, such as buttons versus trackballs. Designers and users should run through different scenarios, identifying potential problems with the interface or generating ideas for new functionality. Mock-ups can also help the designer envision how an interactive system will be incorporated into a physical space (Figure 47.6).

47.4.1.3 Wizard of Oz

Sometimes it is useful to give users the impression that they are working with a real system, even before it exists. Kelley (1983) dubbed this technique the *Wizard of Oz*, based on the scene in the 1939 movie of the same name. The heroine, Dorothy, and her companions ask the mysterious Wizard of Oz for help. When they enter the room, they see an enormous green human head breathing smoke and speaking with a deep, impressive voice. When they return later, they again see the Wizard. This time, Dorothy's small dog pulls back a curtain, revealing a frail old man pulling levers and making the mechanical Wizard of Oz speak. They realize that the impressive being before them is not a wizard at all, but simply an interactive illusion created by the old man.

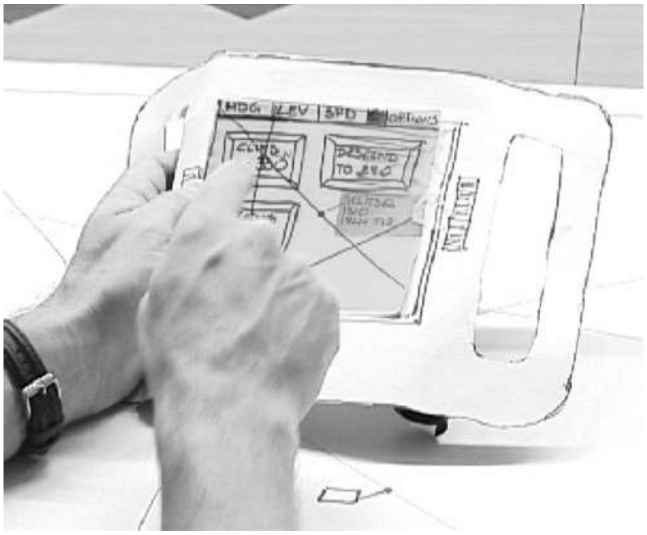


FIGURE 47.5 Mock-up of a hand-held display with carrying handle.

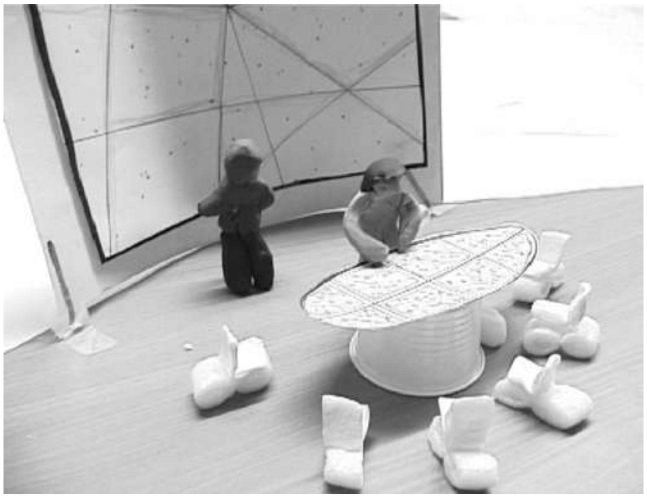


FIGURE 47.6 Scaled mock-up of an air traffic control table, connected to a wall display.

The software version of the Wizard of Oz operates on the same principle. A user sits at a screen and interacts with what appears to be a working program. Hidden elsewhere, the software designer (the Wizard) watches what the user does and, by responding to different user actions, creates the illusion of a working software program. In some cases, the user is unaware that a person, rather than a computer, is operating the system.

The Wizard of Oz technique lets users interact with partially functional computer systems. Whenever they encounter

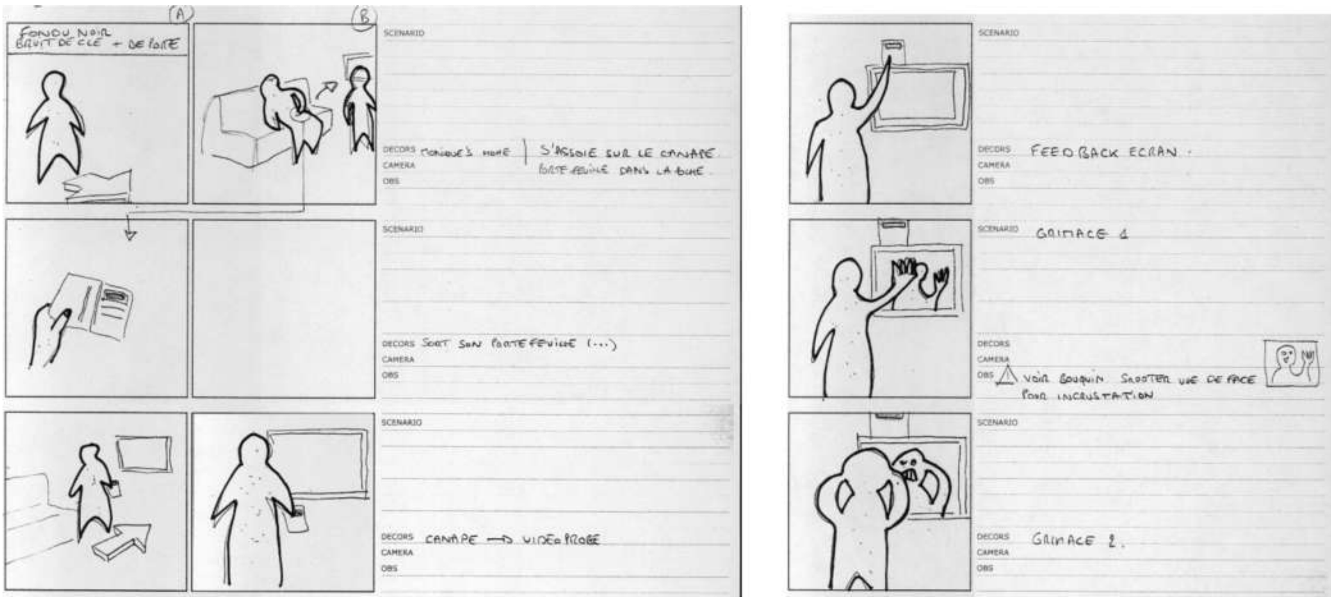

(A)
(B)
FONDU NOIR
BRUIT DE CLÉ + DE PORTE
SCENARIO
DECORS MONIQUE'S HOME | S'ASSOIE SUR LE CANAPE
CAMERA PORTE FEUILLE DANS LA BUHE
OBS
SCENARIO
DECORS SORT SON PORTEFEUILLE (...)
CAMERA
OBS
SCENARIO
DECORS CANAPE → VIDEO PROBE
CAMERA
OBS
SCENARIO
DECORS FEEDBACK ECRAN ·
CAMERA
OBS
SCENARIO GRIMACE 1
DECORS
CAMERA
OBS ⚠ VOIR BOUQUIN SHOOTER UNE DE FACE POUR INCRUSTATION
SCENARIO
DECORS GRIMACE 2.
CAMERA
OBS

FIGURE 47.7 A storyboard for a tangible interface that enables users to establish and manage their connections to a small group of friends and family.

something that has not been implemented (or there is a bug), a human developer who is watching the interaction overrides the prototype system and plays the role destined to eventually be played by the computer. A combination of video and software can work well, depending upon what needs to be simulated.

The Wizard of Oz technique was initially used to develop natural language interfaces (e.g., Chapanis 1982; Good et al. 1984). Since then, the technique has been used in a wide variety of situations, particularly those in which rapid responses from users are not critical. Wizard of Oz simulations may consist of paper prototypes, fully implemented systems, and everything in between.

47.4.1.4 Video Prototyping

Video prototypes (Mackay 1988) use video to illustrate how users will interact with the new system. As explained in an earlier section, they differ from video brainstorming in that the goal is to refine a single design, not generate new ideas. Video prototypes may build upon paper and pencil prototypes and cardboard mock-ups and can use existing software and images of real-world settings.

We begin our video prototyping exercises by reviewing relevant data about users and their work practices, and then review ideas we video brainstormed. The next step is to create a use scenario, describing the user at work. Once the scenario is described in words, the designer develops a storyboard. Similar to a comic book, the storyboard shows a sequence of rough sketches of each action or event, with accompanying actions and/or dialogue (or subtitles), with related annotations that explain what is happening in the scene or the type of shot (Figure 47.7). A paragraph of text in a scenario corresponds to about a page of a storyboard.

Storyboards help designers refine their ideas, generate "what-if" scenarios for different approaches to a story, and communicate with the other people who are involved in the design process. Some storyboards may be informal sketches of ideas, with only partial information. Others follow a predefined format and are used to direct the production and editing of a video prototype. Designers should jot down notes on storyboards as they think through the details of the interaction.

Paper storyboards can be used as is to communicate with other members of the design team. Designers and users can discuss the proposed system and alternative ideas for interacting with it (Figure 47.8). Simple videos of each successive frame, with a voice over to explain what happens, can also be

FIGURE 47.8 Video prototyping. The CPN design team reviews their observations of CPN developers and then discusses several design alternatives. They work out a scenario and storyboard it, then shoot a video prototype that reflects their design.

effective. We usually use storyboards to help us shoot video prototypes, which illustrate how a new system will look to a user in a real-world setting. We find that placing the elements of a storyboard on separate cards and arranging them (Mackay and Pagani 1994) helps the designer experiment with different linear sequences and insert or delete video clips. Continuing to the next step, that is, creating a video prototype based on the storyboard, forces designers to consider the design details in even greater depth.

Storyboards, even very informal ones, are essential guides for shooting video prototypes. To avoid spending time in postproduction, we use a technique called "editing-in-the-camera" (see Mackay 2000) in which each video clip, guided by the storyboard, is shot in the order that it will be viewed. With a well-designed storyboard, this is just as easy and results in a finished video prototype at the end of a one-hour session. Note that today's digital video cameras include editing features in the camera, which introduces the temptation to make editing changes on the fly. Our students who do this consistently take more time than their colleagues who do not, usually with worse results. In general, we recommend avoiding post hoc editing and just following the storyboard.

We use title cards, as in a silent movie, to separate the clips. This both simplifies shooting and makes it easier for the viewer to follow the story. Title cards also provide the only acceptable way to edit while you are shooting: if you make an error, you should address it immediately by rewinding to the last title card and continue shooting from there.

Video prototypes take several forms. In some, a narrator explains each event and several people on the sidelines may be necessary to move images and illustrate the interaction. In others, actors simply perform the movements and the viewer is expected to understand the interaction without a voice over. You can easily create simple special effects. For example, time-lapse photography allows you to have images appear and disappear based on the user's interaction. For example, record a clip of a user pressing a button, press "pause" on the camera, add a pop-up menu, then restart the camera, to create the illusion of immediate system feedback.

Video prototypes should begin with a title, followed by an establishing shot that shows the user in the context defined by the scenario. Next, create a series of close-up and midrange shots, interspersed with title cards to explain the story. Place a final title card with credits at the end. We print blank title cards on colored paper to make it easier to search for the sections of the video later: When you fast-forward through video, a solid blue or red frame clearly stands out (for detailed examples of video-based prototyping techniques, see Mackay 2002).

Video prototypes are fixed prototypes. However, it is also possible to use video as an open prototype, in which users can interact with the prototype in an open-ended way. Video thus becomes a tool for sketching and visualizing interactions. For example, we sometimes use a live video camera as a Wizard of Oz tool and capture the interaction with a second video camera. The Wizard should have access to a set of prototyping materials representing screen objects. Other team members stand by ready to help move objects as needed. The live camera is pointed at the Wizard's work area, with either a paper prototype or a partially working software simulation. The resulting image is projected onto a screen or monitor in front of the user. One or more people should be situated so that they can observe the actions of the user and manipulate the projected video image accordingly. This is most effective if the Wizard is well prepared for a variety of events and can present semiautomated information. The user interacts with the objects he or she sees on the screen and the Wizard moves the relevant materials in direct response to each user action. The other camera records the interaction between the user and the simulated software system on the screen, to create either a video brainstorm (for a quick idea) or a fully storyboarded video prototype.

Figure 47.9 shows a Wizard of Oz simulation with a live video camera, video projector, whiteboard, overhead projector, and transparencies. The setup allows two people to experience how they would communicate via a new interactive communication system. One video camera films the blond woman, who can see and talk to the brunette woman. Her image is projected live onto the left side of the wall. An overhead projector displays hand drawn transparencies, manipulated by two other people, in response to gestures made by the brunette woman. The entire interaction is videotaped by a second video camera. Note that participants at a workshop on user interfaces for air-traffic control created this video: none of the participants had ever used video prototyping techniques but they were able to set up this Wizard of Oz style environment and use it to generate new interaction ideas in less than 30 minutes.

Combining Wizard of Oz and video is a particularly powerful prototyping technique because it gives the person playing the user a real sense of what it might actually feel like to interact with the proposed tool, long before it has been implemented. Seeing a video clip of someone else interacting with a simulated tool is more effective than simply hearing about

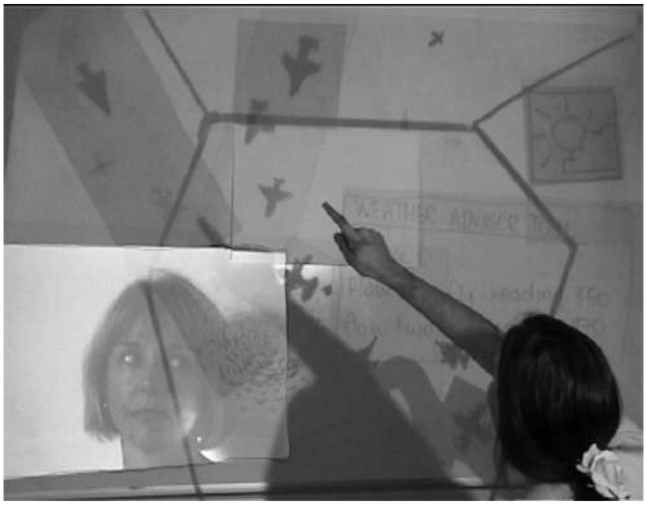


FIGURE 47.9 Complex Wizard of Oz simulation, with projected image from a live video camera and transparencies projected from an overhead projector.

it, but interacting with it directly is more powerful still. Note that video should be used with caution, particularly when video prototypes are taken out of their original design setting (for a more detailed discussion of the ethical issues involved, see Mackay 1995).

Video prototyping may act as a form of specification for developers, enabling them to build the precise interface, both visually and interactively, created by the design team. This is particularly useful when moving from offline to online prototypes, which we now describe.

47.4.2 Online Rapid Prototyping Techniques

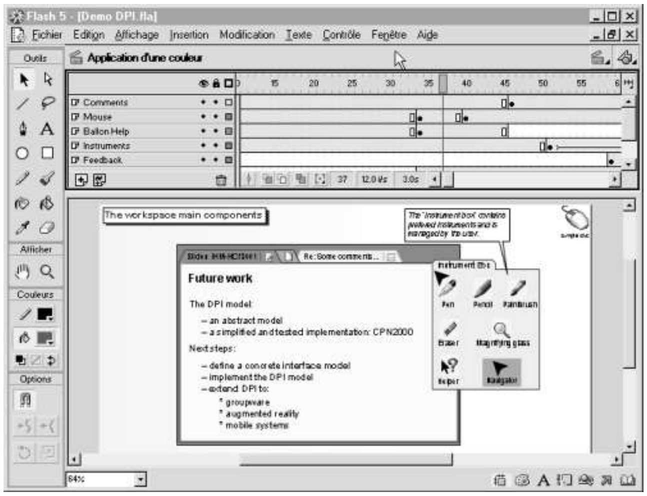
The goal of online rapid prototyping is to create higher precision prototypes than can be achieved with offline techniques. Such prototypes may prove useful to better communicate ideas to clients, managers, developers and end users. They are also useful for the design team to fine tune the details of a layout or an interaction. They may exhibit problems in the design that were not apparent in less precise prototypes. Finally, they may be used early on in the design process for low precision prototypes that would be difficult to create offline, such as when very dynamic interactions or visualizations are needed.

The techniques presented in this section are sorted by interactivity. We start with noninteractive simulations, such as animations, followed by interactive simulations that provide fixed or multiple-path interactions. We finish with scripting languages, which support open interactions.

47.4.2.1 Noninteractive Simulations

A noninteractive simulation is a computer-generated animation that represents what a person would see of the system if he or she were watching over the user's shoulder. Noninteractive simulations are usually created when offline prototypes, including video, fail to capture a particular aspect of the interaction and it is important to have a quick prototype to evaluate the idea. It is usually best to start by creating a storyboard to describe the animation, especially if the developer of the prototype is not a member of the design team.

One of the most widely used tools for noninteractive simulations is Macromedia Director™. The designer defines graphical objects called "sprites," and defines paths along which to animate them. The succession of events, such as when sprites appear and disappear, is determined with a time line. Sprites are usually created with drawing tools, such as Adobe Illustrator or Deneba Canvas, painting tools, such as Adobe Photoshop, or even scanned images. Director is a very powerful tool and experienced developers use it quickly to create sophisticated interactive simulations. (However, it is still faster to create noninteractive simulations.) Other similar tools exist on the market such as Adobe AfterEffects and Macromedia Flash (Figure 47.10).



The workspace main components

The "Instrument box" contains prefixed instruments and is managed by the user.

Future work

The DPI model:

– an abstract model
– a simplified and tested implementation: CPN2000

Next steps:

– define a concrete interface model
– implement the DPI model
– extend DPI to:
* groupware
* augmented reality
* mobile systems

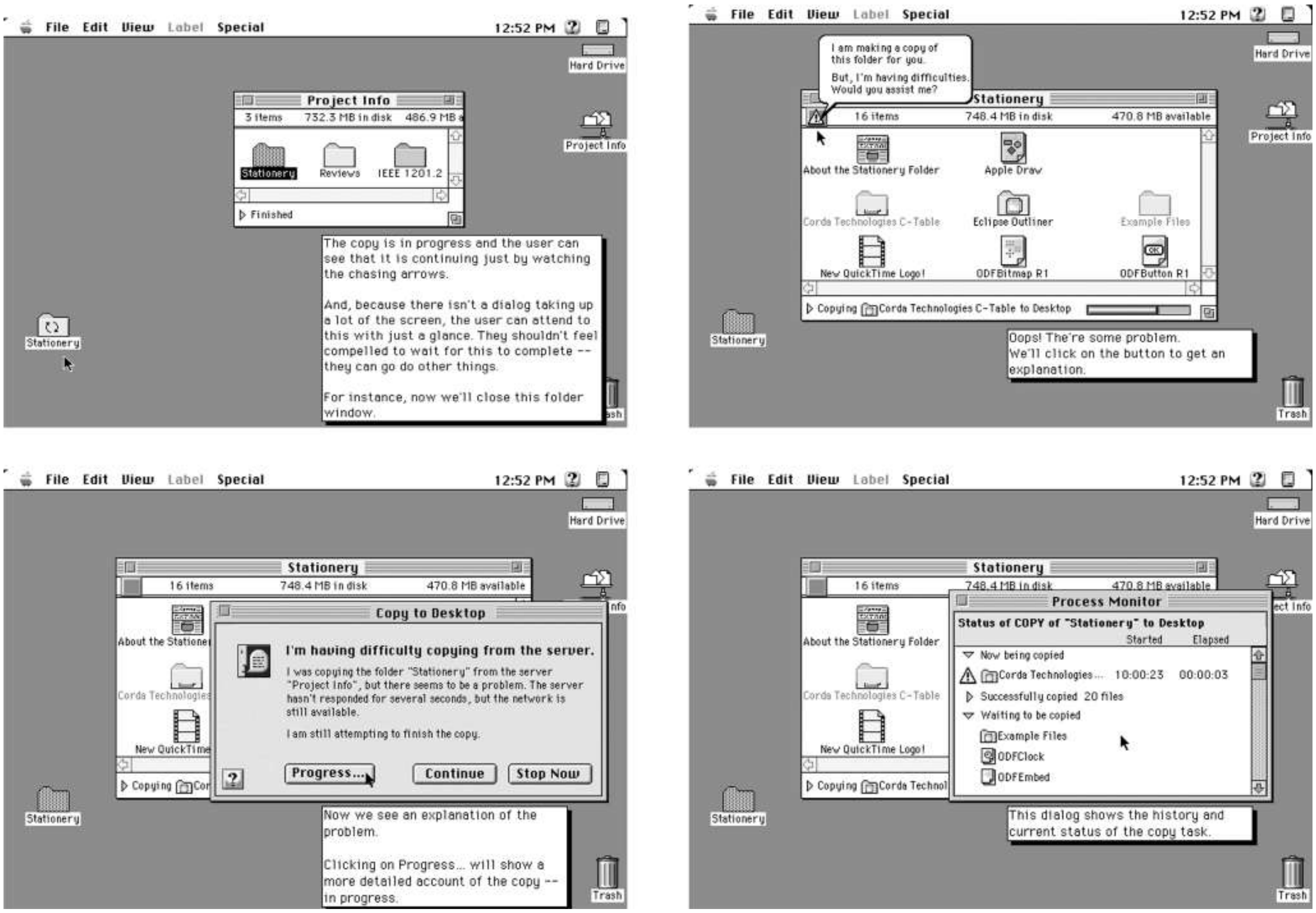
FIGURE 47.10 A noninteractive simulation of a desktop interface created with Macromedia Flash. The time-line (top) displays the active sprites while the main window (bottom) shows the animation. (© O. Beaudoux, with permission)

Figure 47.11 shows a set of animation movies created by Curbow to explore the notion of accountability in computer systems (Dourish 1997). These prototypes explore new ways to inform the user of the progress of a file copy operation. They were created with Macromind Director by combining custom-made sprites with sprites extracted from snapshots of the Macintosh Finder. The simulation features cursor motion, icons being dragged, windows opening and closing, and so forth. The result is a realistic prototype that shows how the interface looks and behaves, that was created in just a few hours. Note that the simulation also features text annotations to explain each step, which helps document the prototype.

Noninteractive animations can be created with any tool that generates images. For example, many web designers use Adobe Photoshop to create simulations of their websites. Photoshop images are composed of various layers that overlap like transparencies. The visibility and relative position of each layer can be controlled independently. Designers can quickly add or delete visual elements, simply by changing the characteristics of the relevant layer. This permits quick comparisons of alternative designs and helps visualize multiple pages that share a common layout or banner. Skilled Photoshop users find this approach much faster than most web-authoring tools.

We used this technique in the CPN2000 project (Mackay, Ratzer, and Janecek 2000) to prototype the use of transparency. After several prototyping sessions with transparencies and overhead projectors, we moved to the computer to understand the differences between the physical transparencies and the transparent effect as it would be rendered on a computer screen. We later developed an interactive prototype with OpenGL, which required an order of magnitude more time to implement than the Photoshop mock-up.

Even a spreadsheet program can be used for prototyping: Berger (2006) described the use of Microsoft Excel to prototype form-based interfaces. First, a template is created that contains a number of reusable elements by taking advantage


File Edit View Label Special
12:52 PM
Hard Drive
Project Info
Stationery
Trash
Project Info
3 items 732.3 MB in disk 486.9 MB a
Stationery Reviews IEEE 1201.2
Finished
Stationery
16 items 748.4 MB in disk 470.8 MB available
About the Stationery Folder
Apple Draw
Corda Technologies C-Table
Eclipse Outliner
Example Files
New QuickTime Logo!
ODFBitmap R1
ODFButton R1
Copying Corda Technologies C-Table to Desktop
Copy to Desktop
Progress...
Continue
Stop Now
Process Monitor

The copy is in progress and the user can see that it is continuing just by watching the chasing arrows.

And, because there isn't a dialog taking up a lot of the screen, the user can attend to this with just a glance. They shouldn't feel compelled to wait for this to complete -- they can go do other things.

For instance, now we'll close this folder window.

I am making a copy of this folder for you.

But, I'm having difficulties. Would you assist me?

Oops! The're some problem. We'll click on the button to get an explanation.

I'm having difficulty copying from the server.

I was copying the folder "Stationery" from the server "Project Info", but there seems to be a problem. The server hasn't responded for several seconds, but the network is still available.

I am still attempting to finish the copy.

Now we see an explanation of the problem.

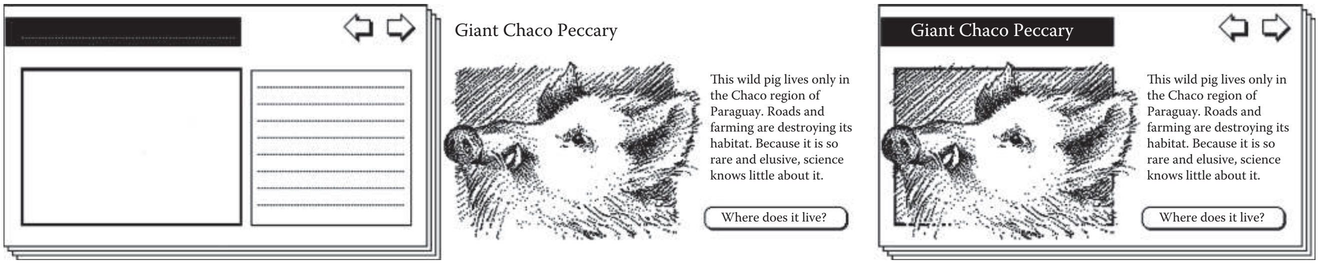
Clicking on Progress... will show a more detailed account of the copy -- in progress.

Status of COPY of "Stationery" to Desktop

```
                                   Started    Elapsed
▽ Now being copied
⚠ Corda Technologies...            10:00:23   00:00:03
▷ Successfully copied  20 files
▽ Waiting to be copied
   Example Files
   ODFClock
   ODFEmbed
```

This dialog shows the history and current status of the copy task.

FIGURE 47.11 Frames from an animated simulation created with Macromind Director. (© D. Curbow, with permission.)



This wild pig lives only in the Chaco region of Paraguay. Roads and farming are destroying its habitat. Because it is so rare and elusive, science knows little about it.

This wild pig lives only in the Chaco region of Paraguay. Roads and farming are destroying its habitat. Because it is so rare and elusive, science knows little about it.

FIGURE 47.12 A Hypercard card (right) is the combination of a background (left) and the card's content (center). (From Apple Computer. 1996. *Programmer's Guide to MacApp*. Cupertino, CA: Apple Computer, Inc. With permission.)

of the workbook feature of Excel, where multiple pages can be presented with a tabbed interface. Then, prototypes are created by copying and pasting items from the template into a blank page, taking advantage of the table structure of the spreadsheet to create grid layouts.

47.4.2.2 Interactive Simulations

Designers can also use tools such as Adobe Photoshop to create Wizard of Oz simulations. For example, the effect of dragging an icon with the mouse can be obtained by placing the icon of a file in one layer and the icon of the cursor in another layer and by moving either or both layers. The visibility of layers, as well as other attributes, can also create more complex effects. Like Wizard of Oz and other paper prototyping techniques, the behavior of the interface is generated by the user who is operating the Photoshop interface.

More specialized tools, such as Hypercard and Macromedia Director, can be used to create simulations with which the user can directly interact. Hypercard (Goodman 1987) was one of the most successful early prototyping tools. It was an authoring environment based on a stack metaphor: a stack contains a set of cards that share a background, including fields and buttons. Each card can also have its own unique contents, including fields and buttons (Figure 47.12). Using a scripting language, stacks, cards, fields, and buttons can react to user events, such as clicking a button, as well as system events, for example, when a new card is displayed or about to disappear.

Interfaces can be prototyped quickly with this approach, by drawing different states in successive cards and using buttons to switch from one card to the next. Multiple-path interactions can be programmed by using several buttons on each card. Interactions that are more open require advanced use of the scripting language, but are easy to master with a little practice.

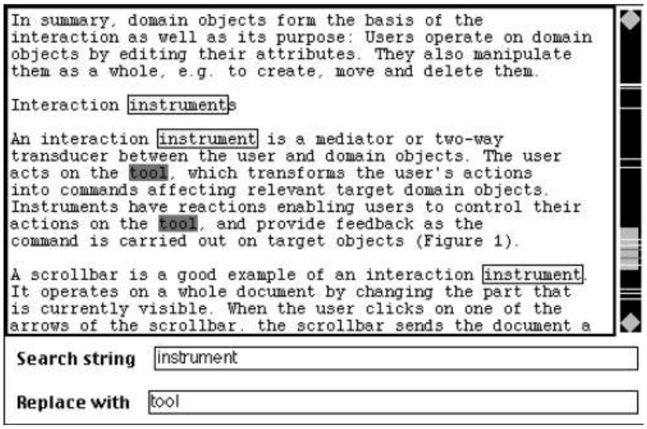
Macromind Director uses a different metaphor, attaching behaviors to sprites and to frames of the animation. For example, a button can be defined by attaching a behavior to the sprite representing that button. When the sprite is clicked, the animation jumps to a different sequence. This is usually coupled with a behavior attached to the frame containing the button that loops the animation on the same frame. As a result, nothing happens until the user clicks the button, at which point the animation skips to a sequence where, for example, a dialogue box opens. The same technique can be used to make the OK and Cancel buttons of the dialogue box interactive. Typically, the Cancel button would skip to the original frame while the OK button would skip to a third sequence. Director comes with a large library of behaviors to describe such interactions, so that prototypes can be created completely interactively. New behaviors can also be defined with a scripting language called Lingo.

47.4.2.3 Scripting Languages

Scripting languages are the most advanced rapid prototyping tools. As with the interactive-simulation tools described above, the distinction between rapid prototyping tools and development tools is not always clear. Scripting languages make it easy to quickly develop throwaway prototypes (a few hours to a few days), which may or may not be used in the final system, for performance or other technical reasons.

A scripting language is a programming language that is both lightweight and easy to learn. Most scripting languages are interpreted or semicompiled; for example, the user does not need to go through a compile-link-run cycle each time the script (program) is changed. Scripting languages can be forbidding: they are not strongly typed and nonfatal errors are ignored unless explicitly trapped by the programmer. Scripting languages are often used to write small applications for specific purposes and can serve as glue between preexisting applications or software components.

Tcl (Ousterhout 1994) is particularly suitable to develop user interface prototypes (or small to medium-size applications) because of its Tk user interface toolkit. Tcl was inspired by the syntax of the Unix shell and makes it very easy to interface existing applications by turning the application programming interface (API) into a set of commands that can be called directly from a Tcl script. Tk features all the traditional interactive objects (called "widgets") of a UI toolkit: buttons, menus, scrollbars, lists, dialogue boxes, and so forth. A widget is typically created with only one line. For example,



In summary, domain objects form the basis of the
interaction as well as its purpose: Users operate on domain
objects by editing their attributes. They also manipulate
them as a whole, e.g. to create, move and delete them.

Interaction instruments

An interaction instrument is a mediator or two-way
transducer between the user and domain objects. The user
acts on the tool, which transforms the user's actions
into commands affecting relevant target domain objects.
Instruments have reactions enabling users to control their
actions on the tool, and provide feedback as the
command is carried out on target objects (Figure 1).

A scrollbar is a good example of an interaction instrument.
It operates on a whole document by changing the part that
is currently visible. When the user clicks on one of the
arrows of the scrollbar, the scrollbar sends the document a

Search string: instrument

Replace with: tool

FIGURE 47.13 Using the Tk text and canvas widgets to prototype a novel search and replace interaction technique. (From Beaudouin-Lafon, M. 2000. Instrumental interaction: An interaction model for designing post-WIMP user interfaces. In *Proceedings ACM Human Factors in Computing Systems, CHI'2000.* 446–53. The Hague, Netherlands.)

```
button .dialogbox.ok -text OK -command {destroy
.dialogbox}.
```

This command creates a button, called ".dialogbox.ok," whose label is "OK." It deletes its parent window ".dialogbox" when the button is pressed. A traditional programming language and toolkit would take 5 to 20 lines to create the same button.

Tcl also has two advanced, heavily parameterized widgets: the text widget and the canvas widget. The text widget can be used to prototype text-based interfaces. Any character in the text can react to user input using tags. For example, it is possible to turn a string of characters into a hypertext link. In Beaudouin-Lafon (2000), the text widget was used to prototype a new method for finding and replacing text. When entering the search string, all occurrences of the string are highlighted in the text (Figure 47.13). Once a replace string has been entered, clicking an occurrence replaces it (the highlighting changes from yellow to red). Clicking a replaced occurrence returns it to its original value. This example also uses the canvas widget to create a custom scrollbar that displays the positions and status of the occurrences.

The Tk canvas widget is a drawing surface that can contain arbitrary objects: lines, rectangles, ovals, polygons, text strings, and widgets. Tags allow behaviors (e.g., scripts) that are called when the user acts on these objects. For example, an object that can be dragged will be assigned a tag with three behaviors: button-press, mouse-move, and button-up. Because of the flexibility of the canvas, advanced visualization and interaction techniques can be implemented more quickly and easily than with other tools. For example, Figure 47.14 shows a prototype exploring new ideas to manage overlapping windows on the screen (Beaudouin-Lafon 2001). Windows can be stacked and slightly rotated so that

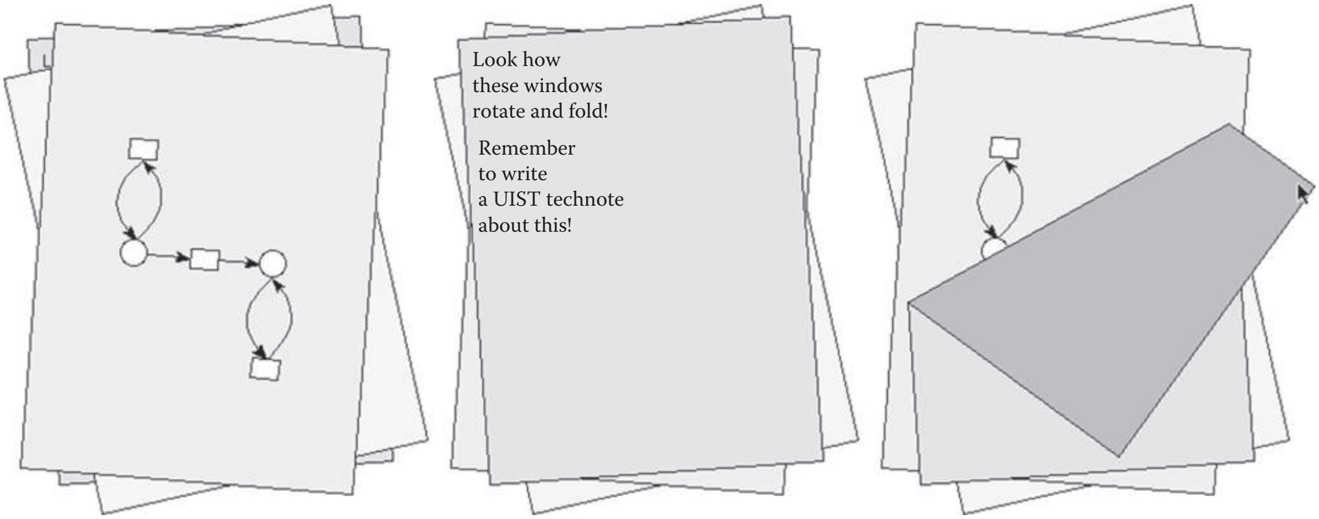


FIGURE 47.14 Using the Tk canvas widget to prototype a novel window manager. (From Beaudouin-Lafon, M. 2001. Novel interaction techniques for overlapping Windows. In *Proceedings of ACM Symposium on User Interface Software and Technology, UIST 2001*, Orlando, Florida. *CHI Letters* 3(2):153–4.)

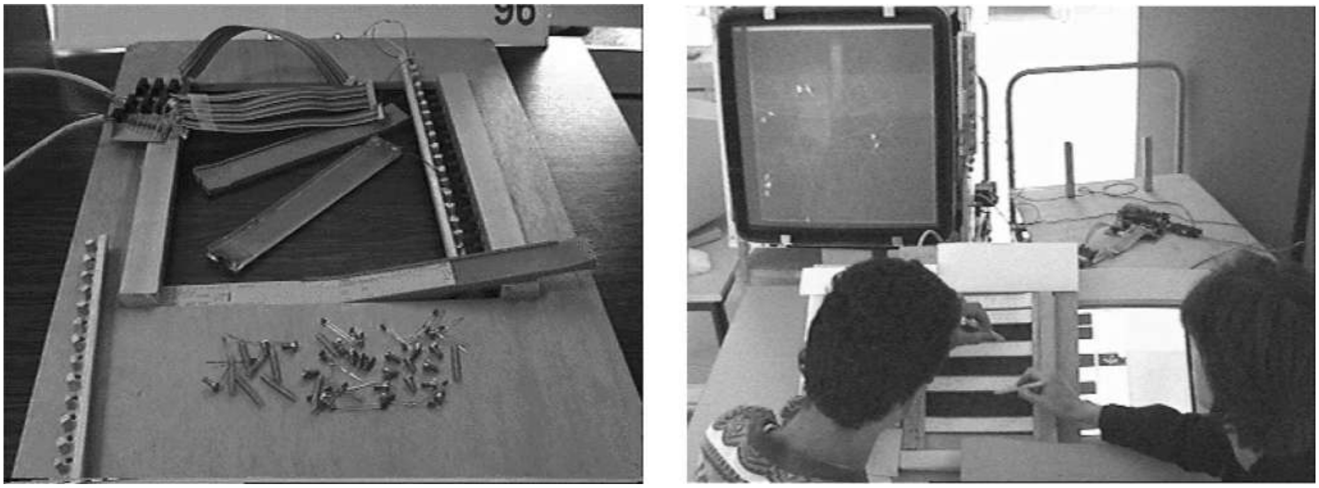


FIGURE 47.15 Caméléon's augmented stripboard (left) is a working hardware prototype that identifies and captures hand-writing from paper flight strips. Members of the design team test the system (right), which combines both hardware and software prototypes into a single interactive simulation.

it is easier to recognize them, and they can be folded so it is possible to see what is underneath without having to move the window. Even though the prototype is not perfect, that is, folding a window that contains text is not properly supported, it was instrumental in identifying a number of problems with the interaction techniques and finding appropriate solutions through iterative design.

Tcl and Tk can also be used with other programming languages. For example, Pad++ (Bederson and Meyer 1998) is implemented as an extension to Tcl/Tk: the zoomable interface is implemented in C for performance and is accessible from Tk as a new widget. This makes it easy to prototype interfaces that use zooming. It is also a way to develop evolutionary prototypes: a first prototype is implemented completely in Tcl, then parts of it are reimplemented in a compiled language to performance. Ultimately, the complete system may be implemented in another language, although it is more likely that some parts will remain in Tcl.

Software prototypes can also be used in conjunction with hardware prototypes. Figure 47.15 shows an example of a hardware prototype that captures handwritten text from a paper flight strip (using a combination of a graphics tablet and a custom-designed system for detecting the position of the paper strip holder). We used Tcl/Tk, in conjunction with C11, to present information on a RADAR screen (tied to an existing air traffic control simulator) and to provide feedback on a touch-sensitive display next to the paper flight strips (Caméléon, Mackay, et al. 1998). The user can write in the ordinary way on the paper flight strip, and the system interprets the gestures according to the location of the writing on the strip. For example, a change in flight level is automatically sent to another controller for confirmation and a physical tap on the strip's ID lights up the corresponding aircraft on the RADAR screen. A later section of this chapter expands this approach to the development of prototypes of mixed reality and pervasive computing systems.

47.5 ITERATIVE AND EVOLUTIONARY PROTOTYPES

Prototypes may also be developed with traditional software development tools. In particular, high-precision prototypes usually require a level of performance that cannot be achieved with the rapid online prototyping techniques previously described. Similarly, evolutionary prototypes intended to evolve into the final product require more traditional software development tools. Finally, even shipped products are not final, since subsequent releases can be viewed as initial designs for prototyping the next release.

Development tools for interactive systems have been in use for over 20 years and are constantly being refined. Several studies have shown that the part of the development cost of an application spent on the user interface is 50% to 80% of the total cost of the project (Myers and Rosson 1992). The goal of development tools is to shift this balance by reducing production and maintenance costs. Another goal of development tools is to anticipate the evolution of the system over successive releases and support iterative design.

The lowest level tools are graphical libraries, which provide hardware independence for painting pixels on a screen and handling user input and window systems that provide an abstraction (the window) to structure the screen into several virtual terminals. User interface toolkits structure an interface as a tree of interactive objects called "widgets," while user interface builders provide an interactive application to create and edit those widget trees. Application frameworks build on toolkits and UI builders to facilitate the creation of typical functions such as cut/copy/paste, undo, help, and interfaces based on editing multiple documents in separate windows. Model-based tools semiautomatically derive an interface from a specification of the domain objects and functions to be supported. Finally, user interface development environments (UIDEs) provide an integrated collection of tools for the development of interactive software.

Before we describe some of these categories in more detail, it is important to understand how they can be used for prototyping. It is not always best to use the highest level available tool for prototyping. High-level tools are most valuable in the long term because they make it easier to maintain the system, port it to various platforms, or localize it to different languages. These issues are irrelevant for vertical and throwaway prototypes, so a high-level tool may prove less effective than a lower level one.

The main disadvantage of higher level tools is that they constrain or stereotype the types of interfaces they can implement. User interface toolkits usually contain a limited set of widgets, and it is expensive to create new ones. If the design must incorporate new interaction techniques, such as bimanual interaction (Kurtenbach et al. 1993) or zoomable interfaces (Bederson and Hollan 1994), a user interface toolkit will hinder rather than help prototype development. Similarly, application frameworks assume a stereotyped application with a menu bar, several toolbars, a set of windows holding documents, and so forth. Such a framework would be inappropriate for developing a game or a multimedia educational CD-ROM that requires a fluid, dynamic, and original user interface.

Finally, developers need to truly master these tools, especially when prototyping in support of a design team. Success depends on the programmer's ability to change the details quickly as well as the overall structure of the prototype. A developer will be more productive when using a familiar tool than if forced to use a more powerful but unknown tool.

Since a complete tour of development tools for interactive systems is beyond the scope of this chapter, we focus on those tools that can be used most effectively for prototyping: user interface toolkits, user interface builders, and user interface development environments.

47.5.1 User Interface Toolkits

User interface toolkits are probably the most widely used tool nowadays to implement applications. All three major platforms (Unix/Linux, MacOS, and Windows) come with at least one standard UI toolkit. The main abstraction provided by a UI toolkit is the widget. A widget is a software object that has three facets that closely match the MVC model (Krasner and Pope 1988): a presentation, a behavior, and an application interface.

The presentation defines the graphical aspect of the widget. The overall presentation of an interface is created by assembling widgets into a tree. Widgets such as buttons are the leaves of the tree. Composite widgets constitute the nodes of the tree and control the layout of their children. The behavior of a widget defines the interaction methods it supports: a button can be pressed, a scrollbar can be scrolled, and a text field can be edited. The application interface defines how a widget communicates the results of the user interaction to the rest of the application. It is usually based on a notification mechanism.

One limitation of widgets is that their behaviors are limited to the widget itself. Interaction techniques that involve multiple widgets, such as drag-and-drop, cannot be supported by the widgets' behaviors alone and require separate support in the UI toolkit. Some interaction techniques, such as toolglasses or magic lenses (Bier et al. 1993), break the widget model both with respect to the presentation and the behavior and cannot be supported by traditional toolkits. In general, prototyping new interaction techniques requires either implementing them within new widget classes, which is not always possible, or not using a toolkit at all. Implementing a new widget class is typically more complicated than implementing the new technique outside the toolkit, for example, with a graphical library, and is rarely justified for prototyping. Many toolkits provide a blank widget, such as the Canvas in Tk or JFrame in Java Swing (Eckstein, Loy, and Wood 1998), which can be used by the application to implement its own presentation and behavior. This is usually a good alternative to implementing a new widget class, even for production code.

User interface toolkits have been an active area of research over the past 15 years. InterViews (Linton, Vlissides, and Calder 1989) inspired many modern toolkits and user interface builders. Some recent research toolkits that can be used for prototyping include SubArctic (Hudson, Mankoff, and Smith 2005) and Satin (Hong and Landay 2000). The latter is dedicated to ink-based interaction and is used with the Silk and Denim UIDEs described below.

A number of toolkits have also shifted away from the widget model to address other aspects of user interaction. For example, GroupKit (Roseman and Greenberg 1996, 1999) was designed for groupware, Jazz (Bederson, Meyer, and Good 2000) for zoomable interfaces, the Visualization (Schroeder, Martin, and Lorensen 1997) and InfoVis (Fekete 2004) toolkits for visualization, Inventor (Strass 1993) for 3D graphics, and Metisse (Chapuis and Roussel 2005) for window management (Figure 47.16).

Creating an application or a prototype with a UI toolkit requires solid knowledge of the toolkit and experience with programming interactive applications. In order to control the complexity of the interrelations between independent pieces of code (creation of widgets, callbacks, global variables, etc.), it is important to use well-known design patterns (Gamma et al. 1995) such as Command, Chain of Responsibility, Mediator, and Observer. Otherwise the code quickly becomes unmanageable and, in the case of a prototype, unsuitable to design space exploration.

47.5.2 User Interface Builders

User interface builders leverage user interface toolkits by allowing the developer of an interactive system to create the presentation of the user interface, such as the tree of widgets, interactively with a graphical editor. The editor typically features a palette of widgets that the user can use to draw the interface in the same way as a graphical editor is used to create diagrams with lines, circles, and rectangles. The presentation attributes of each widget can be edited interactively as well as the overall layout. This saves a lot of time that would otherwise be spent writing and fine tuning rather dull code that creates widgets and specifies their attributes. It also makes it extremely easy to explore and test design alternatives.

Apple's Interface Builder (Figure 47.17) is a descendant of the NeXT interface builder (NeXT Corporation 1991). The palette to the right contains the available widgets. The user interface of the application was created by dragging these widgets into the application window at the top left. The bottom left window contains icons representing the application objects. By dragging connectors between widgets and these objects, a significant part of the behavior of the interface can be created interactively. The user interface can be tested at any time by switching the builder to test mode, making it easy to verify that it behaves as expected. The same application built directly with the underlying toolkit would require dozens of lines of code and significant debugging.

User interface builders are widely used to develop prototypes, as well as final applications. They are easy to use, they make it easy to change the look of the interface, and they hide a lot of the complexity of creating user interfaces with UI toolkits. However, despite their name, they do not cover the whole user interface. Therefore they still require a significant amount of programming, a good knowledge of the underlying toolkit and an understanding of their limits, especially when prototyping novel visualization and interaction techniques.

47.5.3 User Interface Development Environments

A number of high-level tools exist for creating interactive applications. They are often based on user interface toolkits, and they sometimes include an interface builder. These

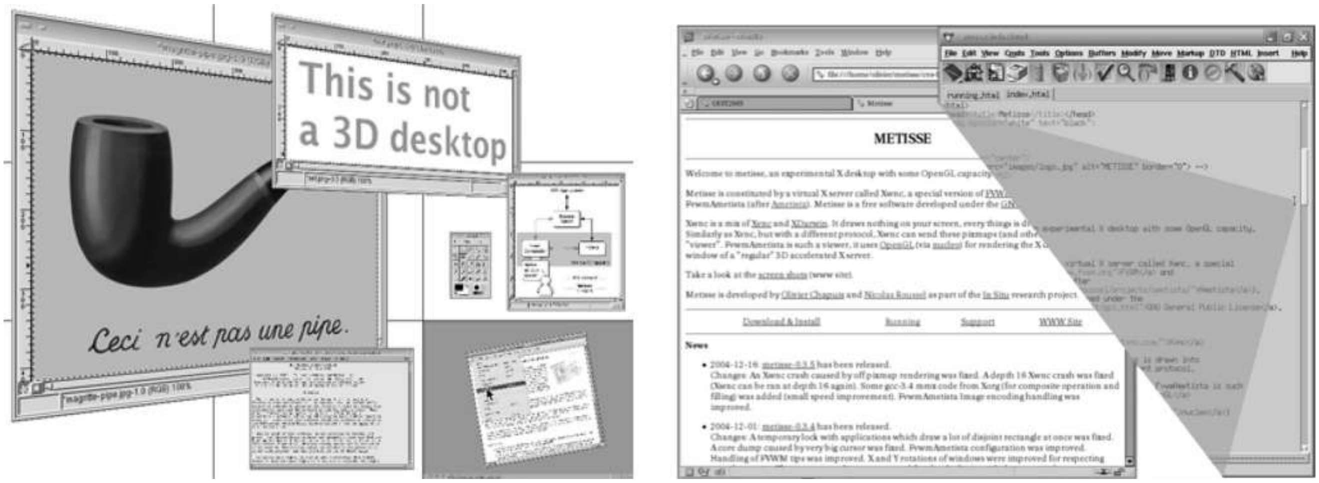

This is not a 3D desktop
Ceci n'est pas une pipe.
File Edit View Go Bookmarks Tools Window Help
File Edit View Cmds Tools Options Buffers Modify Move Markup DTD HTML Insert Help
running.html index.html
METISSE
Download & Install
Running
Support
WWW Site
News

FIGURE 47.16 Two prototypes of a window system implemented with Metisse. The one on the right implements folding windows. Unlike in Figure 47.15, here it works with real applications. (From Chapuis, O., and N. Roussel. 2005. Metisse is not a 3D desktop! In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, UIST '05*, 13–22. Seattle, WA. With permission.)

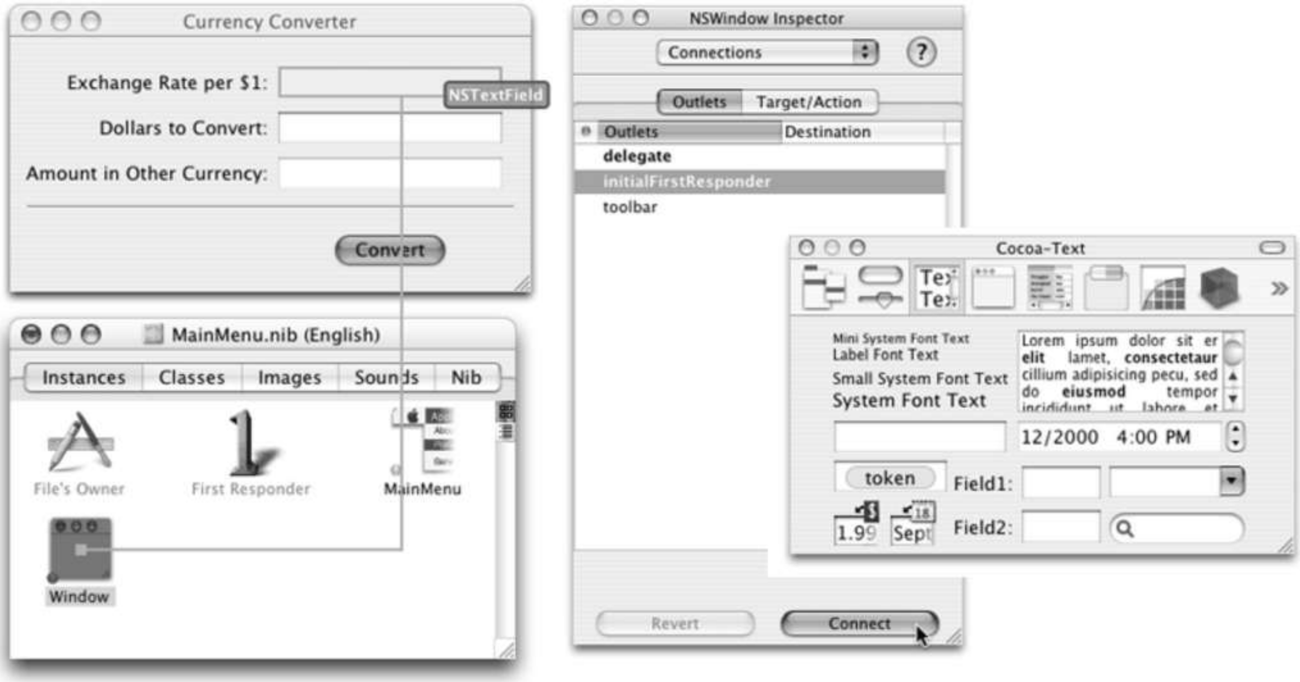


FIGURE 47.17 Interface Builder with the window being built (top-left), application objects (bottom-left), inspector (center), and widget palette (right).

tools are often referred to as "user interface development environments."

The simplest of these tools are application frameworks, which address stereotyped applications. For example, many applications have a standard form where windows represent documents that can be edited with menu commands and tools from palettes; each document may be saved into a disk file; standard functions such as copy/paste, undo, and help are supported. Implementing such stereotyped applications with a UI toolkit or UI builder requires replicating a significant amount of code to implement the general logic of the application and the basics of the standard functions. Application frameworks address this issue by providing a shell that the developer fills in with the functional core and the actual presentation of the nonstandard parts of the interface. Most frameworks have been inspired by MacApp, a framework developed in the 1980s to develop applications for the Macintosh (Apple Computer 1996). Some frameworks are more specialized than MacApp. For example, Unidraw (Vlissides and Linton 1990) is a framework for creating graphical editors in domains, such as technical and artistic drawing, music composition, or circuit design. By addressing a smaller set of applications, such a framework can provide more support and significantly reduce implementation time.

Mastering an application framework takes time. It requires knowledge of the underlying toolkit and the design patterns used in the framework, and a good understanding of the design philosophy of the framework. A framework is useful because it provides a number of functions "for free," but at the same time it constrains the design space that can be explored. Frameworks can prove effective for prototyping if their limits are well understood by the design team.

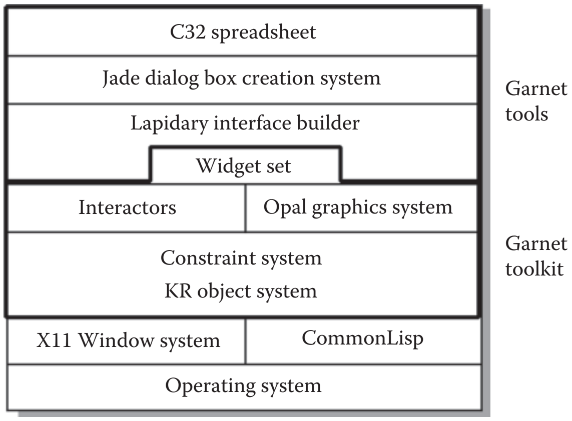


FIGURE 47.18 The Garnet toolkit and tools. (From Myers, B. A., D. A. Giuse, R. B. Dannenberg, B. Vander Zander, D. S. Kosbie, E. Pervin, A. Mickish, and P. Marchal. 1990. Garnet: Comprehensive support for graphical, highly-interactive user interfaces. *IEEE Comput* 23(11):71–85.)

UIDEs consist in assembling a set of tools into an environment where different aspects of an interactive system can be specified and generated separately. For example, Garnet (Figure 47.18) and its successor Amulet (Myers et al. 1990; Myers et al. 1997) provide a comprehensive set of tools, including a traditional user interface builder, a semi-automatic tool for generating dialogue boxes, a user interface builder based on a demonstration approach, and so forth.

Some UIDEs include tools that are specifically designed for prototyping. For example, Silk (Landay and Myers 2001) is

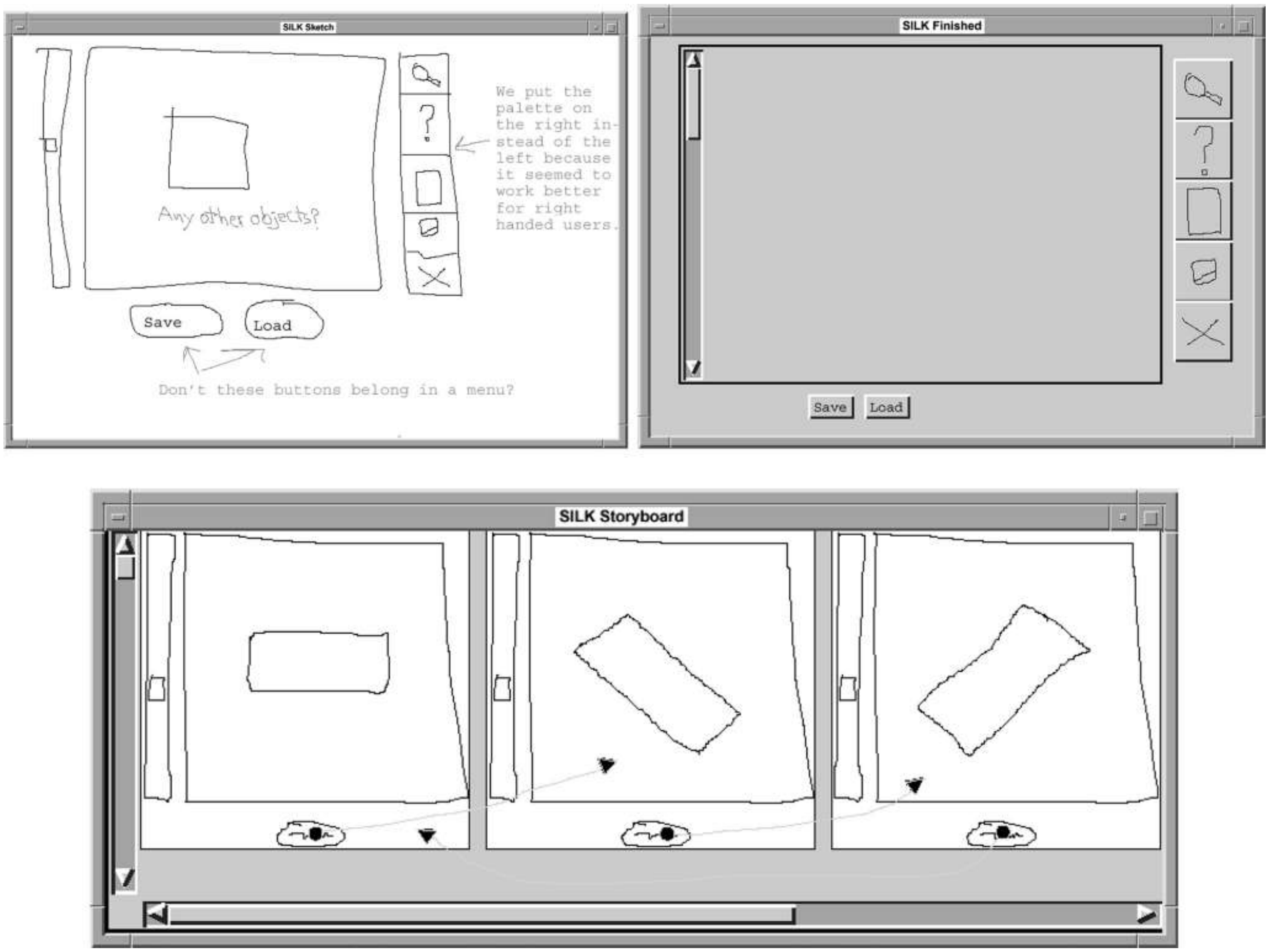

SILK Sketch
We put the palette on the right instead of the left because it seemed to work better for right handed users.
Any other objects?
Save
Load
Don't these buttons belong in a menu?
SILK Finished
Save
Load
SILK Storyboard

FIGURE 47.19 A sketch created with Silk (top left) and its automatic transformation into a Motif user interface (top right). A storyboard (bottom) is used to test sequences of interactions, here a button that rotates an object. (From Landay, J., and B. A. Myers. 2001. Sketching interfaces: Toward more human interface design. *IEEE Comput* 34(3):56–64. With permission.)

a tool aimed at the early stages of design, when interfaces are sketched rather than prototyped in software. The user sketches a user interface directly on the screen (Figure 47.19). Using gesture recognition, Silk interprets the marks as widgets, annotations, and so forth. Even in its sketched form, the user interface is functional: buttons can be pressed, tools can be selected in a toolbar, and so on. The sketch can also be turned into an actual interface, that is, using the Motif toolkit. Finally, storyboards can be created to describe and test sequences of interactions. Monet (Li and Landay 2005) expands this approach by supporting the specification of animations and continuous interaction such as drag-and-drop. Silk and Monet therefore combine some aspects of offline and online prototyping techniques, trying to get the best of both worlds. Another example is Denim (Lin et al. 2000), which addresses the prototyping of websites. Like Silk, it uses a sketch-based interface to allow designers to quickly enter ideas about the overall design of the website as well as individual pages and test the navigation. This illustrates a current trend in research where online tools attempt to support not only the development of the final system, but the whole design process. The following section will also illustrate this trend in the new and very active areas of mixed reality and pervasive computing.

47.6 PROTOTYPING MIXED REALITY AND PERVASIVE COMPUTING SYSTEMS

While most examples in the previous sections concerned traditional graphical user interfaces (GUIs), there is an increasing need to address the design and prototyping of systems that combine the physical and online worlds. The current trend towards pervasive computing and mixed reality started with Weiser's (1991) seminal article on ubiquitous computing and was carried on by augmented reality (Wellner, Mackay, and Gold 1993) and tangible interfaces (Holmquist, Schmidt, and Ullmer 2004). A common theme of these approaches is to combine interaction with the real world and interaction with online information, taking advantage of humans' abilities to interact with physical artifacts.

While mixed reality emphasizes the role of physical objects, pervasive computing emphasizes the role of the physical space. Both raise new and difficult issues when

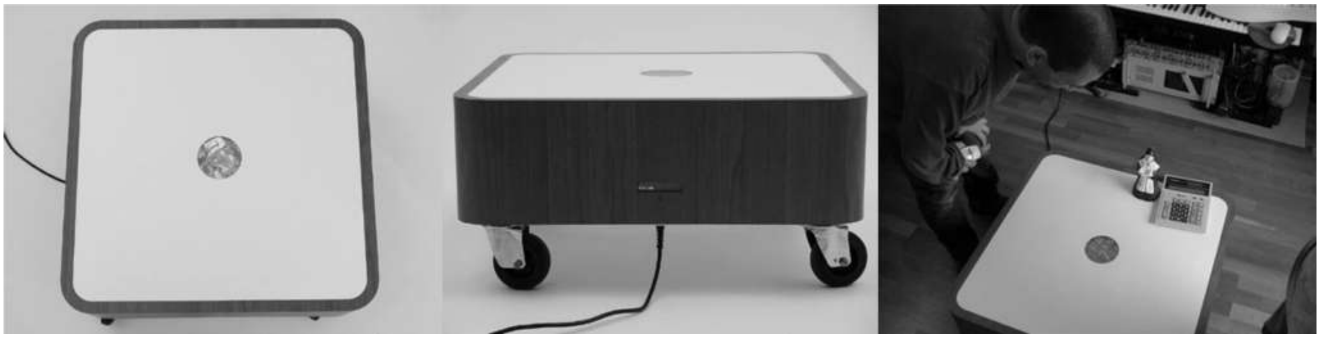


FIGURE 47.20 The drift table. (From Gaver, W., J. Bowers, A. Boucher, H. Gellerson, S. Pennington, A. Schmidt, A. Steed, N. Villar, and B. Walker. 2004. The drift table: Designing for ludic engagement. In *Proc. CHI '04 Design Expo*. New York: ACM Press. With permission.)

trying to actually design and build such systems. At the prototyping stage, they typically require a larger effort than when designing a GUI. First, the range of possible interactions is much broader and typically includes gesture, eye-gaze, or speech. These techniques are already difficult to incorporate into traditional interfaces: in a mixed reality or pervasive computing environment, the user is much less constrained, making recognition and interpretation of users' actions much harder. Designers must then address recognition errors and more generally the effects of context on the sensing techniques. Second, the range of artifacts to design is larger than with desktop interfaces, as it includes the physical artifacts that the users may manipulate; in pervasive computing, the role of user location, the issues of wireless network coverage and sensor range, the difficulty of providing feedback where needed are all new challenges that need to be addressed. Finally, these systems are not necessarily used for tasks where speed of execution and productivity are the primary measures of success. Whether they are used in the home, for tourism, such as tour guides, for assisted living or plain entertainment, the wider and less well-mapped design space typically requires extensive prototyping work.

For example, Boucher and Gaver (2006) described their experiences in designing the drift table, a coffee table with a small screen in the center displaying an aerial view of England and load sensors measuring the weight of objects on the table. According to the measured weight and its estimated location, the aerial view slowly drifts in that direction, as if navigating with a hot air balloon. While the resulting design is conceptually simple, it required many prototypes, both to explore what it should do and how it could be built. The final prototype (Figure 47.20) had to be aesthetically pleasing as well as fully functional for testing in people's homes. This example shows that the higher precision prototypes had to address interaction, function, and looks simultaneously. This proved particularly challenging but is common when prototyping these types of systems.

The prototyping strategies that we have described, such as horizontal, vertical, and scenario-based prototypes, are still valid for prototyping mixed reality and pervasive computing systems, as well as some of the prototyping techniques, such as offline prototyping with mock-ups and video prototyping. Online prototyping, however, requires specific tools and iterative and incremental prototyping, even more so. The rest of this section explores some of the existing tools.

Phidgets (Greenberg and Fitchett 2001) are physical sensors and actuators that can be controlled from a computer through a USB connection. A wide range of sensors is available (light, motion, distance, pressure, humidity, etc.), as well as input devices (buttons, sliders), LEDs, LCD displays, motors, and RFID tag readers. By hiding their implementations and exposing their functionalities through a standard software interface, phidgets can greatly facilitate the design and prototyping of tangible interfaces. For example, they can be embedded into a physical artifact and programmed from the computer to prototype a mobile device. Since all phidgets can be represented on the computer screen, they can also be used for Wizard of Oz settings, where the Wizard can directly control the output phidgets such as LEDs and motors.

Papier-Mâché (Klemmer et al. 2004) is a toolkit for creating tangible interfaces based on vision, RFID tags, and barcodes. It is more adapted to the development of augmented paper applications than Phidgets and provides a richer development environment. The DART toolkit (MacIntyre et al. 2005) takes a similar approach, but targets augmented and mixed reality applications. The development environment is based on Macromedia Director, with the explicit goal of being familiar to designers. In the context of prototyping, both toolkits can be used for applications other than those that they explicitly target. For example, DART can superimpose a computer-generated model with live or recorded video, which can be used to put an image or 3D model of a device being designed in the real world.

Topiary (Li, Hong, and Landay 2004) is a tool for prototyping pervasive applications that use the locations of people, places, and objects. Unlike the above tool, its goal is not to create real-world prototypes allowing to test a system in situ. Rather, it is an online tool that allows to create and test scenarios on the screen, using maps representing the environment and icons representing the people, places, and objects of interest. Once a scenario has been created, the tool allows the user to experience it either from a bird's eye view or from the perspective of a particular user by displaying the user's PDA. Such scenario-based prototyping can be very useful in the early stages of design, when exploring ideas.

Finally, a CAPpella (Dey et al. 2004), is a tool to help end users create or customize context-aware applications. It uses programming by demonstration so that users only have to show examples of sensor data and the system's reaction. While the system targets end users, it can also be used by designers to create prototypes of context-aware applications. By hiding the complexity of programming recognizers, it allows for a fast exploration of alternatives.

In summary, while mixed reality and pervasive computing create new challenges for designers, a number of tools are starting to appear to help with the design process in general and prototyping in particular. Interestingly, these tools can also be used to prototype more traditional applications. For example, phidgets can be used to control an online application in a Wizard of Oz setting, DART can be used to create scenarios that mix videos with models of future artifacts, and a CAPpella can be used to experiment with recognition-based applications. As mixed reality and pervasive computing systems become more widely available, they will no doubt provide more tools to be used by designers in unexpected ways to help with the prototyping of interactive systems at large.

47.7 CONCLUSION

Prototyping is an essential component of interactive system design. Prototypes may take many forms, from rough sketches to detailed working prototypes. They provide concrete representations of design ideas and give designers, users, developers and managers an early glimpse into how the new system will look and feel. Prototypes increase creativity, allow early evaluation of design ideas, help designers think through and solve design problems, and support communication within multidisciplinary design teams.

Prototypes, because they are concrete and not abstract, provide a rich medium for exploring a design space. They suggest alternate design paths and reveal important details about particular design decisions. They force designers to be creative and to articulate their design decisions. Prototypes embody design ideas and encourage designers to confront their differences of opinion. The precise aspects of a prototype offer specific design solutions: designers can then decide to generate and compare alternatives. The imprecise or incomplete aspects of a prototype highlight the areas that must be refined or require additional ideas.

We defined prototypes and then discussed them as design artifacts. We introduced four dimensions by which they can be analyzed: representation, precision, interactivity, and evolution. We then discussed the role of prototyping within the design process and explained the concept of creating, exploring, and modifying a design space. We briefly described techniques for generating new ideas to expand the design space and techniques for choosing among design alternatives to contract the design space.

We described a variety of rapid prototyping techniques for exploring ideas quickly and inexpensively in the early stages of design, including offline techniques (from paper and pencil to video) and online techniques (from fixed to interactive simulations). We then described iterative and evolutionary prototyping techniques for working out the details of the online interaction, including development tools and software environments. Finally, we addressed the emerging fields of mixed reality and pervasive computing and described the new challenges they raise for interactive systems design, as well as some of the tools available for prototyping them.

We view design as an active process of working with a design space, expanding it by generating new ideas and contracting as design choices are made. Prototypes are flexible tools that help designers envision this design space, reflect upon it, and test their design decisions. Prototypes are diverse and can fit within any part of the design process, from the earliest ideas to the final details of the design. Perhaps most important, prototypes provide one of the most effective means for designers to communicate with each other, as well as with users, developers, and managers, throughout the design process.

REFERENCES

Apple Computer. 1996. *Programmer's Guide to MacApp*. Cupertino, CA: Apple Computer, Inc.

Beaudouin-Lafon, M. 2000. Instrumental interaction: An interaction model for designing post-WIMP user interfaces. In *Proceedings ACM Human Factors in Computing Systems, CHI '2000*. 446–53. The Hague, Netherlands.

Beaudouin-Lafon, M. 2001. Novel interaction techniques for overlapping Windows. In *Proceedings of ACM Symposium on User Interface Software and Technology, UIST 2001*, Orlando, Florida. *CHI Letters* 3(2):153–4.

Beaudouin-Lafon, M. 2004. Designing interaction, not interfaces. In *Proceedings of the Working Conference on Advanced Visual interfaces, AVI '04*, 15–22. Gallipoli, Italy.

Beaudouin-Lafon, M., and M. Lassen. 2000. The architecture and implementation of a post-WIMP graphical application. In *Proceedings of ACM Symposium on User Interface Software and Technology, UIST 2000,* San Diego, CA. *CHI Lett* 2(2):181–90.

Beaudouin-Lafon, M., and W. Mackay. 2000. Reification, polymorphism and reuse: Three principles for designing visual interfaces. In *Proceedings Conference on Advanced Visual Interfaces, AVI 2000*, 102–9. Palermo, Italy.

Beck, K. 2000. *Extreme Programing Explained*. New York: Addison-Wesley.

Bederson, B. B., and J. Hollan. 1994. Pad++: A zooming graphical interface for exploring alternate interface physics. In *Proceedings of ACM Symposium on User Interface Software and Technology, UIST '94*, 17–26. Marina del Rey.

Bederson, B. B., and J. Meyer, 1998. Implementing a zooming interface: Experience building Pad++. *Softw Pract Exp* 28(10): 1101–35.

Bederson, B. B., J. Meyer, and L. Good. 2000. Jazz: An extensible zoomable user interface graphics toolkit in Java. In *Proceedings of ACM Symposium on User Interface Software and Technology, UIST 2000*, San Diego, CA. *CHI Lett* 2(2): 171–80.

Berger, N. 2006. The excel story. *Interactions* 13(1):14–7.

Bier, E., M. Stone, K. Pier, W. Buxton, and T. De Rose. 1993. Toolglass and magic lenses: The see-through interface. In *Proceedings ACM SIGGRAPH*, 73–80. Anaheim, CA.

Bødker, S. 1999. Scenarios in user-centered design: Setting the stage for reflection and action. In *Proceedings of the 32nd Annual Hawaii International Conference on System Sciences, HICSS-32*, Vol 3, article 3053, 11. Wailea, HI: IEEE Computer Society.

Bødker, S., P. Ehn, J. Knudsen, M. Kyng, and K. Madsen. 1988. Computer support for cooperative design. In *Proceedings of the CSCW '88 ACM Conference on Computer-Supported Cooperative Work*, 377–93. Portland, OR.

Boehm, B. 1988. A spiral model of software development and enhancement. *IEEE Comput* 21(5):61–72.

Boucher, A., and W. Gaver. 2006. Developing the drift table. *Interactions* 13(1):24–7.

Chapanis, A. 1982. Man/Computer research at Johns Hopkins. In *Information Technology and Psychology: Prospects for the Future*, ed. R. A. Kasschau, R. Lachman, and K. R. Laughery, 238–49. New York, NY: Praeger Publishers, Third Houston Symposium.

Chapuis, O., and N. Roussel. 2005. Metisse is not a 3D desktop! In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, UIST '05*, 13–22. Seattle, WA.

Collaros, P. A., and L. R. Anderson. 1969. Effect of perceived expertness upon creativity of members of brainstorming groups. *J Appl Psychol* 53:159–163.

de Vreede, G. J., R. Briggs, R. van Duin, and B. Enserink. 2000. Athletics in electronic brainstorming: Asynchronous brainstorming in very large groups. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, HICSS-33,* Vol 1, article 1042, 11. Wailea, HI: IEEE Computer Society.

Dey, A. K., R. Hamid, C. Beckmann, I. Li, and D. Hsu. 2004. A CAPpella: Programming by demonstration of context-aware applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04*, 33–40. Vienna, Austria.

Diehl, M., and W. Stroebe. 1987. Productivity loss in brainstorming groups: Towards the solution of a riddle. *J Personal Soc Psychol,* 53(3):497–509. Washington, DC: American Psycology Association.

Dijkstra-Erikson, E., W. E. Mackay, and J. Arnowitz. 2001. Trialogue on design of. *ACM/Interact* 109–17.

Dourish, P. 1997. Accounting for system behaviour: Representation, reflection and resourceful action. In *Computers and Design in Context*, ed. M. Kyng and L. Mathiassen, 145–70. Cambridge, MA: MIT Press.

Eckstein, R., M. Loy, and D. Wood. 1998. *Java Swing*. Cambridge, MA: O'Reilly.

Fekete, J. D. 2004. The InfoVis toolkit. In *Proceedings of the 10th IEEE Symposium on Information Visualization, InfoVis '04,* 167–74. Austin, Texas: IEEE Press.

Frishberg, N. 2006. Prototyping with junk. *Interactions* 13(1):21–3.

Gamma, E., R. Helm, R. Johnson, and J. Vlissides. 1995. *Design patterns, elements of reusable object-oriented software*. Reading, MA: Addison-Wesley.

Gaver, W., J. Bowers, A. Boucher, H. Gellerson, S. Pennington, A. Schmidt, A. Steed, N. Villar, and B. Walker. (2004). The Drift Table: Designing for ludic engagement. Proc. CHI '04 Design Expo. New York: ACM Press.

Good, M. D., J. A. Whiteside, D. R. Wixon, and S. J. Jones. 1984. Building a user-derived interface. *Commun ACM* 27(10):1032–43. New York: ACM.

Goodman, D. 1987. *The Complete HyperCard Handbook*. New York: Bantam Books.

Greenbaum, J., and M. Kyng, eds. 1991. *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Greenberg, S., and C. Fitchett. 2001. Phidgets: Easy development of physical interfaces through physical widgets. In *Proceedings of the 14th Annual ACM Symposium on User interface Software and Technology, UIST '01*, 209–18. Orlando, FL.

Holmquist, L. E., A. Schmidt, and B. Ullmer. 2004. Tangible interfaces in perspective. *Pers Ubiquitous Comput* 8(5):291–3. Heidelberg: Springer.

Hong, J. I., and J. A. Landay. 2000. SATIN: A toolkit for informal ink-based applications. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology, UIST '00*, 63–72. San Diego, CA.

Houde, S., and C. Hill 1997. What do prototypes prototype? In *Handbook of Human Computer Interaction*, 2nd ed., ed. M. G. Helaner, T. K. Landauer, and P. V. Pradhu, 367–81. North-Holland.

Hudson, S. E., J. Mankoff, and I. Smith. 2005. Extensible input handling in the subArctic toolkit. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '05*, 381–90. Portland, Oregon.

Kelley, J. F. 1983. An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of CHI '83 Conference on Human Factors in Computing Systems*, 193–6. Boston, MA.

Klemmer, S. R., J. Li, J. Lin, and J. A. Landay. 2004. Papier-mache: Toolkit support for tangible input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04*, 399–406. Vienne, Austria.

Krasner, E. G., and S. T. Pope. 1988. A cookbook for using the model-view-controller user interface paradigm in small-talk-80. *J Object-Oriented Program,* 27–49.

Kurtenbach, G., G. Fitzmaurice, T. Baudel, and W. Buxton. 1993. The design of a GUI paradigm based on tablets, two-hands, and transparency. In *Proceedings of ACM Human Factors in Computing Systems, CHI '97*, 35–42. Atlanta, GA.

Landay, J., and B. A. Myers. 2001. Sketching interfaces: Toward more human interface design. *IEEE Comput* 34(3):56–64.

Li, Y., J. I. Hong, and J. A. Landay. 2004. Topiary: A tool for prototyping location-enhanced applications. In *Proceedings of the 17th Annual ACM Symposium on User interface Software and Technology, UIST '04*, 217–26. Santa Fe, NM.

Li, Y., and J. A. Landay. 2005. Informal prototyping of continuous graphical interactions by demonstration. In *Proceedings of the 18th Annual ACM Symposium on User interface Software and Technology, UIST '05*, 221–30. Seattle, WA.

Lin, J., M. W. Newman, J. I. Hong, and J. A. Landay. 2000. DENIM: Finding a tighter fit between tools and practice for website design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '00*, 510–17. Hague, Netherlands.

Linton, M. A., J. M. Vlissides, and P. R. Calder. 1989. Composing user interfaces with InterViews. *IEEE Comput* 22(2):8–22.

Mackay, W. E. 1988. Video prototyping: A technique for developing hypermedia systems. In *Conference Companion of ACM CHI '88, Conference on Human Factors in Computing*. Washington DC. http://www.lri.fr/~mackay/publications.html (accessed April 2, 2007).

Mackay, W. E. 1995. Ethics, lies and videotape. In *Proceedings ACM Human Factors in Computing Systems, CHI '95*, 138–45. Denver, CO.

Mackay, W. E. 2000. Video techniques for participatory design: Observation, brainstorming & prototyping. In *Tutorial Notes, CHI 2000, Human Factors in Computing Systems.* Hague, Netherlands. http://www.lri.fr/,mackay/publications.html (accessed April 2, 2007).

Mackay, W. E. 2002. Using video to support interaction design, DVD tutorial, INRIA and ACM/SIGCHI. http://stream.cc.gt.atl.ga.us/hccvideos/viddesign.php (accessed April 2, 2007).

Mackay, W. E., and A. L. Fayard. 1997. HCI, natural science and design: A framework for triangulation across disciplines. In *Proceedings of ACM DIS '97, Designing Interactive Systems*, 223–34. Pays-Bas, Amsterdam.

Mackay, W. E., and D. Pagani. 1994. Video mosaic: Laying out time in a physical space. In *Proceedings of ACM Multimedia '94*, 165–72. San Francisco, CA.

Mackay, W. E., A. L. Fayard, L. Frobert, and L. Médini. 1998. Reinventing the familiar: Exploring an augmented reality design space for air traffic Control. In *Proceedings of ACM CHI '98 Human Factors in computing Systems*, 558–65. Los Angeles, CA.

Mackay, W. E., A. Ratzer, and P. Janecek. 2000. Video artifacts for design: Bridging the gap between abstraction and detail. In *Proceedings ACM Conference on Designing Interactive Systems, DIS 2000*, 72–82. New York.

MacIntyre, B., M. Gandy, S. Dow, and J. D. Bolter. 2005. DART: A toolkit for rapid design exploration of augmented reality experiences. *ACM Trans Graph* 24(3):932.

Muller, M. J. 1991. PICTIVE: An exploration in participatory design. In *Proceedigs of ACM CHI '91 Human Factors in Computing Systems*, 225–31. New Orleans, LA.

Myers, B. A., D. A. Giuse, R. B. Dannenberg, B. Vander Zander, D. S. Kosbie, E. Pervin, A. Mickish, and P. Marchal. 1990. Garnet: Comprehensive support for graphical, highly-interactive user interfaces. *IEEE Comput* 23(11):71–85.

Myers, B. A., R. G. McDaniel, R. C. Miller, A. S. Ferrency, A. Faulring, B. D. Kyle, A. Mickish, A. Klimovitski, and P. Doane. 1997. The Amulet environment. *IEEE Trans Softw Eng* 23(6):347–65.

Myers, B. A., and M. B. Rosson. 1992. Survey on user interface programming. In *ACM Conference on Human Factors in Computing Systems, CHI '92*, 195–202. Monterey, CA.

NeXT Corporation. 1991. *NeXT Interface Builder Reference Manual*. Redwood City, CA: NeXT Corporation.

Norman, D. A., and S. W. Draper, eds. 1986. *User Centered System Design*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Osborn, A. 1957. *Applied Imagination: Principles and Procedures of Creative Thinking*. Rev. ed. New York: Scribner's.

Ousterhout, J. K. 1994. *Tcl and the Tk Toolkit*. Reading, MA: Addison Wesley.

Perkins, R., D. S. Keller, and F. Ludolph. 1997. Inventing the Lisa user interface. *ACM Interact* 4(1):40–53.

Rettig, M. 1994. Prototyping for tiny fingers. *Commun ACM* 37(4):21–7.

Raskin, J. 2000. *The Humane Interface*. New York: Addison-Wesley.

Roseman, M., and S. Greenberg. 1996. Building real-time groupware with GroupKit, a groupware toolkit. *ACM Trans Comput Hum Interact* 3(1):66–106.

Roseman, M., and S. Greenberg. 1999. Groupware toolkits for synchronous work. In *Computer-Supported Co-operative Work: Trends in Software Series*, ed. M. Beaudouin-Lafon, 135–68. Chichester: Wiley.

Schroeder, W., K. Martin, and B. Lorensen. 1997. *The Visualization Toolkit*. Upper Saddle River, NJ: Prentice Hall.

Snyder, C. 2003. *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*. San Francisco, CA: Morgan Kaufmann.

Strass, P. 1993. IRIS inventor, a 3D graphics toolkit. In *Proceedings ACM Conference on Object-Oriented Programming, Systems, Languages and Applications, OOPSLA '93*, 192–200. Washington, DC.

Vlissides, J. M., and M. A. Linton. 1990. Unidraw: A framework for building domain-specific graphical editors. *ACM Trans Inf Syst* 8(3):237–68.

Weiser, M. 1991. The computer for the 21st century. *Sci Am* 265(3):94–104.

Wellner, P., W. E. Mackay, and R. Gold, eds. 1993. Special Issue on Computer-Augmented Environments. *Commun ACM*. New York: ACM.

48 Scenario-Based Design

Mary Beth Rosson and John M. Carroll

CONTENTS

48.1 INTRODUCTION

Scenario-based design (SBD) is a family of techniques in which the *use* of a future system is concretely described at an early point in the development process. Narrative descriptions of envisioned usage episodes—user interaction scenarios—are then used in various ways to guide the development of the system that will enable these use experiences.

Like other user-centered approaches, SBD changes the focus of design work from defining system operations (i.e., functional specification) to describing how people will use a system to accomplish work tasks and other activities. However, unlike approaches that analyze and address human behavior and experience through the formal analysis and modeling of well-specified tasks, SBD offers relatively lightweight methods for envisioning future use possibilities.

A user interaction scenario is a *sketch of use*. It is intended to vividly capture the essence of an interaction design, much as a two-dimension paper-and-pencil sketch captures the essence of a physical design. Like any story, a scenario consists of a setting, or situation state, one or more actors with personal motivations, knowledge and capabilities, and various tools and objects that the actors encounter and manipulate. The narrative describes a sequence of actions and events that lead to an outcome. These actions and events are related in a usage context that includes the goals, plans, and reactions of the people taking part in the episode.

Table 48.1 presents a brief scenario in which Jerry, an internal business training professional, is preparing for an upcoming 1-week training session with colleagues at another company location. In the scenario, Jerry has the goal to update his knowledge about the software professionals he will be working with; in the narrative, he uses their microblogs to gather recent updates about these individuals in a relatively unobtrusive fashion (Zhao and Rosson 2009). The scenario describes one way that microblogging might support Jerry's goals. In this sense, it is one potential "solution" to his desire to re-acquaint himself with these remote colleagues; it might be contrasted to other scenarios, for example, more direct information exchanges based on individual e-mails that request updates from everyone or a group update request e-mailed out by the managers.

Designers can quickly construct scenarios like these to make envisioned possibilities more concrete. The example conveys how the social media paradigm of microblogging might be applied to the management of business relationships in distributed work settings. Importantly, the scenario does not propose options as an abstract model or even as a list of features or functions. Instead, it presents

TABLE 48.1
Potential Design Scenario Meeting the Needs of Business Trip Preparation

Jerry Visits a Remote Company Site to Carry Out a Training Session
Jerry works as a trainer in the Human Resources Department of VisionWay, a high-tech software development firm. Next week he will visit the company's Seattle lab to conduct training sessions on a new version management tool recently mandated by upper management for all sites. He has prepared a standard training presentation but wants to refresh his memory of the 30 or so staff members he will be working with; he last visited about 6 months ago.
Jerry knows that many of the tech-savvy staff members in Seattle microblog on a regular basis, so he decides to spend some time every day reviewing their posts. By doing this, he discovers that one group is struggling with a color matcher module in their visual analytics package; by following the link in the code, he is able to grab and insert example modules and method calls from this part of their code library into his training materials. Along the way, he also learns that several of the team members have just joined a softball team and that Susan (one of the managers) has recently had a baby.
During his visit, the training sessions go very well, as the developers discuss how the new version management features will help them manage their current updates. In fact, they get so caught up in their discussion that they turn his training session into an impromptu problem-solving session for three of the most problematic methods.

TABLE 48.2
Concerns Stemming from the Solution-First Approach to Design and Aspects of Scenario-Based Design That Address Each Concern

Hazards of Solution-First Approach	How SBD Can Help
Designers want to select a solution approach quickly, which may lead to premature commitment to their first design ideas.	Because they are concrete but rough, scenarios support visible progress, but relax commitment to the ideas expressed in the scenarios.
Designers attempt to quickly simplify the problem space with external constraints, such as the reuse of familiar solutions.	Because they emphasize people and their experiences, scenarios direct attention to the use-appropriateness of design ideas.
Designers are intent on elaborating their current design proposal, resulting in inadequate analysis of other ideas or alternatives.	Because they are evocative and by nature are incomplete, scenarios promote empathy and raise usage questions at many levels.

these ideas implicitly, embeded within a concrete episode of human–computer interaction (HCI) and personal experience.

User interaction scenarios can be successively detailed to discover and address finer-grained design issues. The scenario in Table 48.1 narrates a story about Jerry's experiences, but does so at a relatively high level of abstraction, focusing on his goals and general behavior, rather than details of user interaction. But even when narrated in this general fashion, scenarios serve as grist for group brainstorming to develop further alternatives or to raise questions about the assumptions behind the scenarios. They can be used to analyze software requirements, as a partial specification of functionality, and can be refined to consider and guide the design of user interface layouts and controls. They can also be used to identify and plan evaluation tasks for usability tests.

48.2 WHY SCENARIO-BASED DESIGN?

One reason that scenarios have become so popular in interactive system design is that they enable rapid communication about usage possibilities and concerns among many different stakeholders. It is easy to write a simple text scenario and takes only a little more effort to enrich it with a rough sketch or storyboard. When designers are working through ideas, they want to make progress quickly, so that they can obtain feedback and continue to refine their ideas. Scenarios are one way to do this.

The design of an interactive system is an ill-defined problem. Such problems tend to evoke a problem-solving strategy termed *solution-first* (Cross 2001). In the solution-first strategy, designers generate and analyze a candidate solution as a means of clarifying the problem state, the allowable moves, and the goal. They exploit the concreteness of their own solution proposals to evoke further requirements for analysis.

A solution-first approach to design is energizing, effective, and efficient; it explains the popularity of contemporary system development approaches like rapid prototyping (Wasserman and Shewmaker 1982) and extreme programming (Beck 1999). But this general strategy also entrains well-known hazards (Cross 2001): Designers tend to generate solutions too quickly, before they analyze what is already known about the problem and possible moves. Once an approach is envisioned, they may have trouble abandoning it when it is no longer appropriate. Designers may too readily reuse pieces of a solution they have used earlier, one that is familiar and accessible, but perhaps not appropriate. They may not analyze their own solutions very well or they may consider too few alternatives when exploring the problem space. In the next three sections, we consider how SBD may help to minimize these hazards of solution-first problem solving (see Table 48.2).

48.2.1 Scenarios Are Concrete but Rough

Design analysis is always indeterminate, because the act of design changes the world within which people act and experience. Requirements always change (Brooks 1995). When designs incorporate technologies that are evolving rapidly, requirements change even more rapidly. The more successful, the more widely adopted a design is, and the more impact it has, the less possible it would have been to determine its correct design requirements. And in any case, refinements in software technology and new perceived opportunities and requirements propel a new generation of designs every 2–3 years.

Design representations that are at once concrete but flexible help to manage ambiguous and dynamic situations. Analysts must be concrete to avoid being swallowed by indeterminacy; they must be flexible to avoid being captured by false steps. Systematic decomposition is a traditional approach to managing ambiguity, but it does not promote flexibility. Instead, designers end up with a set of concrete sub-solutions, each of which is fully specified. Unfortunately, by the time the set of sub-solutions is specified, the requirements often would have changed.

User interaction scenarios reconcile concreteness and flexibility. A scenario envisions a concrete design solution, but it can be couched at many levels of detail. Initially, a scenario may be extremely rough. It specifies a possible design by indicating the tasks users may carry out, but without committing to lower-level details describing *how* the tasks will be carried out or *how* the system will present the functionality for the tasks. The scenario in Table 48.1 is at an intermediate level, with some sense of task flow, but few details about specific user–system interactions.

Concrete material is interpreted more easily and more thoroughly than abstract material. For example, people remember a prototypical example far better than they remember the abstract category to which it belongs (Medin and Schaffer 1978; Rosch et al. 1976). Incomplete material tends to be elaborated with respect to personal knowledge when it is encountered. This process of elaboration creates more robust and accessible memories, relative to memories for more complete material (Wertheimer 1938). The combination of concreteness and incompleteness in scenarios engages a powerful variety of constructive cognitive processes.

The fluidity of design situations demands provisional solutions, tentative commitments. Yet, if every design decision is suspended, the result will be a design space, not a design. A scenario is a concrete design proposal that a designer can evaluate and refine, but it is also rough, so that it can be easily altered, and many details can be deferred.

48.2.2 Scenarios Maintain an Orientation to People and Their Needs

Designers need constraints; there are just too many things that might be designed. The current state of technology development makes some solutions impossible and others irresistible: On the one hand, designers cannot use technology that does not yet exist. On the other hand, designers are caught up in a technological zeitgeist that biases them toward making use of the latest gadgets and gizmos. They are likely to be biased toward familiar technologies, even when they are aware of limitations in these technologies.

Scenarios are work-oriented design objects. They describe systems in terms of the work that users will try to do when they use those systems, ensuring that design will remain focused on the needs and concerns of users (Carroll and Rosson 1990). Scenarios address what has been called the "representational bias" in human cognition—people overestimate the relevance of things that are familiar to them (Kahneman and Tversky 1972; Tversky and Kahneman 1974). For instance, designers who have years of experience with general-purpose e-mail communication mechanisms might see update requests sent out over e-mail listservs as a solution to Jerry's visit preparation needs, even though the stakeholders might be better served by emerging social media like Twitter (http://twitter.com). It is difficult to move beyond the familiar, but generating and sharing a vivid representation of exceptions to the status quo can promote innovative thinking. Scenarios that describe unusual but critical circumstances can provide such a perspective.

The reuse of familiar ideas is just one type of constraint that designers may apply in their solution-first problem solving. Other constraints may arise from the organizational structures within which the design work is embedded. Design projects are often chartered with a priori commitments to follow a systematic decomposition process. This makes them easy to manage, but unlikely to succeed with respect to discovering the real requirements of users and clients. Schedules and resources are often assigned in ways that create on-going conflicts between system designers and usability engineers. Usability engineers need to evaluate scenarios and prototypes at every stage of system development, but if schedules and resources do not provide for this, this work can conflict with software construction and refinement.

Constraints such as these can distract designers with ancillary factors so that they lose sight of what is essential in the design project, namely, the needs and concerns of users. The designer can become "unsituated" with respect to the real design situation, which is not the marketing manager's projections, or the instructional designer's list of steps, or the software engineer's system decomposition. The real design situation is the situation that will be experienced by the user, and the designers need to stay focused on that.

Scenarios can be made even more effective as work-oriented design objects when users are directly involved in creating them. Ackoff (1979) argues that the indeterminacy of design situations makes it imperative that *all* stakeholders participate directly. Scenarios support a simple and natural process of participatory design, where prospective users begin by enacting or relating episodes of current activities, then work iteratively with designers to transform and enrich these scenarios with the opportunities provided by new technologies (Carroll et al. 2000; Chin, Rosson, and Carroll 1997).

48.2.3 Scenarios Are Evocative, Raising Questions at Many Levels

There is a fundamental tension between thinking and doing: thinking impedes progress in doing, and doing obstructs thinking. Sometimes this conflict is quite sharp, as when one must stop and think before taking another step. But frequently it is more a matter of trading off priorities. Designers are intelligent people performing complex and open-ended tasks. They want to reflect on their activities, and they routinely do reflect on their activities. However, people take pride not only in what they know and learn, but in what they can do and in what they actually produce.

Donald Schön (1983, 1987) discusses this conflict extensively in his books on reflective practice. For example, he analyzes a coach reacting to an architecture student's design concept for a school building, which included a spiral ramp intended to maintain openness while breaking up lines of sight (she calls the idea "a Guggenheim"):

> … when I visited open schools, the one thing they complained about was the warehouse quality—of being able to see for miles. It [the ramp] would visually and acoustically break up the volume. (Schön 1987, page 129)

In this episode, the coach feels that the student needs to explore and develop her concept more thoroughly, noting that a ramp has thickness and that this will limit her plans to use the space underneath the ramp; he urges her to draw sections. However, he does not justify this advice; as Schön puts it, he does not reveal "the meanings underlying his questions" (Schön 1987, page 132). Schön regards this as a hopeless confrontation in which no progress can be made on the particular design project or on the larger project of understanding how to design. Both the student and the coach are willing to act publicly and to share actions, but they do not reflect enough on their own and one another's values and objectives and on their interpersonal dynamics.

Reflection is not always comfortable; it entails consideration of one's own competence, thereby opening up the possibility of recognizing one's inadequacies or mistakes. Nonetheless, designers regularly create many opportunities for reflection, for example, organizing design, reviewing meetings, or building prototypes that are used in formative evaluations. Such activities promote identification and integration of different perspectives; they raise concrete and detailed design issues to guide further work. In this way they help designers to reflect on the work they have already done. But they do not evoke reflection *in the context of doing design*. Design reviews and formative evaluations are ancillary activities that must be coordinated with the design itself.

Scenarios help designers to reflect about their ideas in the context of doing design. The narrative is written to evoke an image of people doing things, pursuing goals, using technology in support of these goals. The story enables author and readers to empathize with the people in the situation; this in turn provokes questions about motivations, intentions, reactions, and satisfaction. For example, in the example scenario from Table 48.1, is it valuable for Jerry to discover a specific code issue of concern to the employees? What is the effect of reviewing the discussion of visual analytics in the midst of other unrelated posts? How important is it to also discover the more personal information, for example, regarding the company softball team or the manager's new baby?

Scenarios promote reflection and analysis in part because the human mind is adept at overloading meaning in narrative structures, both in generation and interpretation, as illustrated by the remarkable examples of dreams (Freud 1900), myths (Lévi-Strauss 1967), and folktales (Propp 1958). It is well known that when people communicate, they rely on the *given-new contract* (Haviland and Clark 1974): they assume or allude to relevant background information, then present what is novel. This normative structure eases both the generation and interpretation of narratives.

Schön (1983) describes design as a "conversation" with a situation comprised of many interdependent elements. The designer makes moves and then "listens" to the design situation to understand their consequences:

> "In the designer's conversation with the materials of his design, he can never make a move that has only the effects intended for it. His materials are continually talking back to him, causing him to apprehend unanticipated problems and potentials." (page 101)

When a move produces unexpected consequences, and particularly when it produces undesirable consequences, the designer articulates "the theory implicit in the move, criticizes it, restructures it, and tests the new theory by inventing a move consistent with it" (page 155).

Scenarios often include implicit information about design consequences. Considering again the scenario in Table 48.1, if the microblog solution will rely on a popular social media service like Twitter, the participating employees would need to do no additional work to create information useful to Jerry. At the same time, the use of Twitter also means that Jerry will be able to see whatever a given employee chooses to tweet, whether related to work activities or not. If instead a new tool is created (e.g., a company-hosted microblog), employees may feel they are being asked to do extra work to share their updates; the posts are more likely to provide useful information about current project activity; and Jerry is less likely to encounter personal or social updates. Furthermore, microblogging in general can introduce less positive consequences; with 30 employees to visit, Jerry may find it too tedious to scan the many posts that they have generated. These tradeoffs are important to the scenarios, but often it is enough to imply them (this is an aspect of the roughness property discussed earlier).

There are times, however, when it is useful to reflect more systematically on such tradeoffs by making them explicit. If Jerry finds that Twitter logs for the 30 people are simply too long or diverse to browse and he gives up on his goal, the scenario ends in failure. As a designer, it is important to consider when and how such failure scenarios may occur. In SBD, designers analyze and record the important tradeoffs in each scenario under development, so that they can understand, address, and monitor both the desirable and the undesirable consequences of proposed design moves.

Table 48.3 illustrates how the tradeoffs implied by a scenario (including those that arise through "what if?" reasoning)

TABLE 48.3
Tradeoff Analysis of the Scenario from Table 48.1

Browsing remote colleagues' microblogs prior to making a business visit...
+ Increases the common ground between the visitor and the remote colleagues
+ Leverages any current blogging practices of the employees who are to be visited, along with the microblog reviewing practices of the person planning the visit
− But employees who do not like to microblog, or who do so infrequently, may be ignored
− But the number of posts may be large, with considerable content that is irrelevant

can be captured through *claims analysis*. Features of a scenario are extracted and analyzed with respect to hypothesized positive consequences (upsides, shown in the table with a +) and negative consequences (downsides, shown with a –). Each feature and its upsides and downsides constitutes a *claim*. Depending on the level of specification in the scenario, the features that are analyzed may be either general in nature (e.g., mentioning microblogs as a general technique) or more specific (e.g., emphasizing the formatting of Twitter feed displays). As the analysis takes place, it is important to consider variations of the scenario to develop as balanced a view as possible of the consequences for users. For instance, while the upsides in Table 48.3 capture some of the basic motivation for reviewing microblogs, the downsides are revealed by asking questions like "what if some of the employees never participate in blogging?" and "what if all employees blog many times each day?".

Scenarios and the tradeoffs they imply can help designers move within a design space along a more deliberated path, namely away from or toward specific user consequences (Carroll and Rosson 1991). For example, a design team might decide to leverage employees' pre-existing Twitter practices rather than building a new tool, and instead design a mechanism for summarization to address the information overload that comes from digesting the many small and diverse personal posts. Alternatively, the team may decide that a better solution will be to design a private company-specific microblog service, because it will be more effective at evoking detailed and work-relevant posts, while at the same time recognizing that this may minimize posts with personal content.

Scenarios of use are multifarious design objects; they can describe designs at multiple levels of detail and with respect to multiple perspectives. In this way, they can help designers reflect on several aspects of a problem situation simultaneously. The scenario in Table 48.1 offers a high-level task view, but any scenario can be elaborated to include the moment-to-moment thoughts of its actors to provide a detailed account of their cognitive processing or to explore their individual actions and reactions to convey a detailed functional view. They might also be elaborated in terms of the hardware or software components needed for implementing the envisioned functionality (Rosson and Carroll 1995; Wirfs-Brock 1995). Each of these variations in resolution and perspective is a permutation of a single underlying scenario. The permutations are integrated through their roles as complementary views of the same design object.

Using scenarios in this way makes them a more powerful design representation. They allow the designer the flexibility to develop and analyze key usage scenarios in great detail, for example, to describe core application functionality while merely sketching less critical scenarios. At the same time, designers are able to switch among multiple perspectives, for example, directly integrating usability views with software views. Such a flexible and integrative design object can help designers manage the many interdependent consequences implied by design moves (Rosson and Carroll 2000a).

48.2.4 Potential Pitfalls in Scenario-Based Design

While scenario envisionment and analysis can address many of the concerns of solution-first design, a design process centered on scenarios raises its own tradeoffs. The concreteness and work-orientation that make scenarios effective as user-centered design representations may also introduce their own bias into the process, as designers come to identify with and optimize their ideas for particular actors or activity contexts—the very characteristics that make a story realistic and evocative may also lead designers to adopt too narrow a view. The analysis of consequences implied by a scenario (as in Table 48.3) helps to create a more balanced view of current design ideas, but at the same time it may lead to a focus on incremental changes to their ideas (e.g., to address a perceived downside) rather than to consider more innovative or transformational ideas. Also, like any design process that focuses on largely textual representations, a scenario-based process (with its associated analysis and illustration) can produce an unwelcome documentation burden, as designers may find themselves maintaining and evolving a large set of design scenarios at varying stages of refinement. In the next two sections we consider frameworks, design tools, and other methods that are aimed at reaping the benefits of SBD while also helping to manage costs such as these.

48.3 SCENARIO-BASED DEVELOPMENT

The concrete and work-oriented character of scenarios makes them an effective representation for human-centered design activities, particularly when these activities include participation by end-users or other stakeholders (Carroll et al. 2000; Chin, Rosson, and Carroll 1997; Muller 1992; Muller et al. 1995; Karat 1995; Karat and Bennett 1991; Rosson and Carroll 2002). Scenarios can be quickly developed, shared, and revised; they are easily enriched with sketches, storyboards, or other mock-ups (Erickson 1995; Kyng 1995). A scenario of use can be directed at many concerns in system development, including documentation design and object-oriented software design (Carroll 1995; Carroll 2000). Given these many virtues, it is no surprise that scenarios are pervasive in software design and development (Rosson, Maass, and Kellogg 1988; Weidenhaupt et al. 1998). In this section, we expand on the view that scenarios are a user-centered design representation, introducing a programmatic framework for employing scenarios of use in interactive system design (Carroll 2000; for a comprehensive presentation of these SBD methods, see Rosson and Carroll 2002).

The framework summarized in this section incorporates scenario-based analysis and design into all phases of system development, from problem analysis through usability evaluation and iterative development. We build on the general rationale for SBD described in Section 48.3, but at the same time show how to make the impacts of scenario-based reasoning comprehensive and systematic. The overall process is one of usability engineering, where the scenarios support continual assessment and elaboration of the system's usefulness, ease of use, and user satisfaction. The aim is to develop a rich understanding of current activities and work practices and to use this understanding as a basis for activity transformation.

Figure 48.1 provides an overview of the scenario-based development framework. We assume that system development begins with an initial business case, technology exploration vision or charter, even though the design team's "plan" at this point may be quite sketchy and non-binding. The starting vision motivates a period of intense analysis during which the current situation is examined for problems and opportunities that might be addressed by available technologies. The analysts' understanding of the current situation is synthesized and communicated in problem scenarios and claims. Problem scenarios describe prototypical stakeholders engaged in meaningful activities; the claims enumerate features of the current situation that are understood to have important consequences—both positive and negative—for the scenario actors.

Problem scenarios are transformed and elaborated through several phases of iterative design. Design envisionment is inspired by both metaphors and technology features, while also constrained by knowledge of interactive system design. Each set of scenarios is complemented by claims that analyze the possible positive and negative consequences of key design features. Claims analysis leads designers to reflect on the usage implications of their design ideas while the ideas are being developed.

SBD is mediated by evaluation throughout development. Each narrative serves as a test case for analytic evaluation; each claim hypothesizes usability outcomes for one or more test cases. As they become concrete enough to convey to prospective users, design scenarios are evaluated more directly in empirical usability studies. In these the set of claims are used for mediated evaluation, wherein the hypothesized usage impacts are operationalized and tested explicitly (Scriven 1967). The empirical findings are interpreted with respect to the ongoing claims analysis, refining or redirecting the design efforts. We turn now to a brief example illustrating the key elements of the framework.

48.3.1 Problem Analysis

A challenge for any software development project is identifying the complete and correct set of requirements (Brooks 1995). Many system requirements are functional, addressed by the actual services and information provided by the final system. Other requirements are nonfunctional, for example, the measured quality of the software implementation or user interactions, or pragmatic features of the system development process like schedule, cost, or delivery platform (Rosson and Carroll 2000b; Sommerville 1992; Sutcliffe and Minocha 1998). In SBD, we express an initial high-level view of requirements as a *root concept* (Table 48.4). The root concept enumerates key aspects of the team's starting vision; it is used to guide further analysis and elaboration of system requirements.

Table 48.4 contains a root concept for the business use of microblogs, the design example that we will use to illustrate the SBD framework (Rosson and Carroll 2002). The starting vision and rationale in this case are quite intuitive: there are

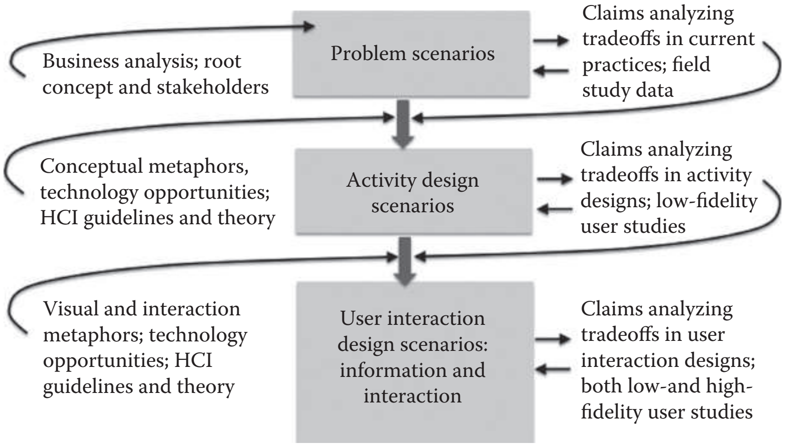

Problem scenarios
Business analysis; root concept and stakeholders
Claims analyzing tradeoffs in current practices; field study data
Activity design scenarios
Conceptual metaphors, technology opportunities; HCI guidelines and theory
Claims analyzing tradeoffs in activity designs; low-fidelity user studies
User interaction design scenarios: information and interaction
Visual and interaction metaphors; technology opportunities; HCI guidelines and theory
Claims analyzing tradeoffs in user interaction designs; both low-and high-fidelity user studies

FIGURE 48.1 An overview of the scenario-based design framework. Scenarios serve as a central representation throughout the development cycle, first describing the goals and concerns of current practices in the problem domain that are successively transformed and refined through an iterative design and evaluation process.

TABLE 48.4
Root Concept for a Microblog Service Intended to Support Workgroup Updating

Component	Contribution to Root Concept
High-level vision	Employees microblog regular small updates about work-relevant activities
Basic rationale	Effort of microblog is small; a browsable and shareable activity log is created as a side effect of the posting activity
Stakeholder goals	
Team members	Short messages easy to create, read/post does not distract
Remote colleagues	Easy to "look in on," increase awareness of remote colleagues
Team managers	Monitor reactions, concerns, interactions among employees
Starting assumptions	Leverage existing practices with social media; incremental and participatory development process

advantages to staying aware of colleagues' activities, especially when these are related to one's own business goals, and the cost of creating or reading any single microblog item is relatively small. As a side effect, the blogs are archived over time, potentially creating a long-term record that can be used in support of other tasks.

The root concept also documents the design team's shared beliefs about the project's major stakeholders. A *stakeholder* is any person or organization who will be affected (either positively or negatively) by the system (Checkland 1981; Muller 1991). It is important to take a broad view of stakeholders, particularly early in problem analysis, so that appropriate individuals and groups will be represented in the analysis activities. In the example, we consider several different classes of employees likely to be affected by the microblog, so that we can consider their distinct goals with respect to system use—an employee may enjoy posting successes and failures, particularly if it provokes help or consultation from peers. Remote employees will appreciate an ability to "peek in" on project groups or individuals of interest (as in Jerry's case). Management may find the posts useful in detecting team issues in advance, developing summaries for group efforts, and so on.

Although the emphasis of SBD is on analysis and iterative development of useful and usable functionality, a range of nonfunctional concerns may also constrain development. These are documented as starting assumptions in the root concept. In our example, we assume that the microblog tool will attempt to leverage any existing habits the employees have with social media (e.g., Facebook, Twitter) and that it will be developed through a participatory process that engages all stakeholder groups.

The root concept sets up a framework for the designer to investigate current practices related to workplace updates and ongoing awareness. This might involve fieldwork, for example, observations of a workgroup for some period of time that documents the different forms of informal communication taking place; it might also appeal to existing empirical works that have documented workplace behavior patterns (Kraut et al. 1993). At times the analysis may be quite modest, perhaps a quick survey of several workgroups aimed at eliciting descriptions of information-sharing activities or a series of semi-structured interviews with both co-located and remote team members and their managers. Important complementary sources of information during this process are any tools or other artifacts used in an activity—for instance a form used to document weekly progress, regular e-mails that are sent from manager to team members or vice versa, specific tools used for filing or sharing project information, and so on. Such artifacts are an excellent source of implicit information about stakeholders' current values and activities (Bødker 1991; Norman 1988; Rosson and Carroll 2002).

Field studies of current practices generate rich and diverse data about the specific needs and opportunities associated with the stakeholders identified in the root concept. To direct these data systematically toward design efforts, a more abstract representation of themes and relations is useful. For example, an affinity diagram (the analysis team posts and organizes individual observations; Beyer and Holtzblatt 1998) can be helpful in discovering general themes. Other useful techniques include diagrams of the stakeholder relationships, hierarchical task analysis of central activities, and summaries of how collected artifacts support group activities (Rosson and Carroll 2002; Chapter 2).

In SBD, the prime outcome of problem analysis is a set of problem scenarios and claims. Each problem scenario is a narrative of current practices that offers a synthetic view of the actors, themes, relationships, and artifacts discovered in the fieldwork. These scenarios are not design-neutral, however. Even during early analysis activities, the team holds some vision of how technology might enhance current practice. The fieldwork and related analyses will inevitably be colored by this vision (Carroll et al. 1994). If the team fails to establish a vision or creates inconsistent or contradictory visions, this too will influence requirements analysis, but in a less positive fashion.

An effective technique for generating problem scenarios is to first describe a set of hypothetical stakeholders—individuals who represent a synthesis of the people studied during the fieldwork, perhaps supplemented by demographic or market analysis data. It is important to create a rich but realistic view of these individuals, because they will form the basis for describing and later transforming current activities and experiences (see the discussion section for how these descriptions relate to the use of personas). As we continue the example, we focus on both Jerry, the company employee who is planning a training visit, and Susan, the manager of one of the groups he will be training.

TABLE 48.5
Problem Scenario Describing How Jerry Currently Uses Twitter to Prepare for His Visits

Problem Scenario: Jerry Visits a Remote Company Site to Carry Out a Training Session

Jerry is a training professional who has worked for human resources in VisionWay for 15 years. Over the years, he has met and gotten to know many VisionWay employees distributed at the company's five locations and always looks forward to meeting with them as part of the training sessions he leads. Next week, he will be conducting a session at their northeast lab for three software development groups. One of them is managed by Susan, a hard-working and talented young woman who is quickly moving up the technical management chain.

Before any training session, Jerry likes to find out what the participants have been working on lately, so that he can quickly establish common ground and use their current work as examples in his training. He knows that many VisionWay employees use Twitter on a regular basis, so he decides to review their Twitter feeds instead of sending out a bunch of e-mails. After quite a bit of scanning, he locates a couple of brief posts from two of Susan's group members expressing frustration with some software; he infers that they are managing consistency within their visual analytics tool. Along the way, he also discovers that some of the staff joined a co-ed softball team; he enjoys their light-hearted teasing of other teams. In Susan's blog, he sees that she just had a baby and is now only returning to work; he reads on to hear about her experiences at work-baby balancing and looks at baby pictures. In the end, he has just a vague sense of groups' technical issues, not enough to work up any examples for his training materials.

During his visit, the training sessions go well, because he spends time at the beginning to find out more about the versioning problem as well as problems plaguing the other two groups. He does his best to adapt his examples on the fly; he teases the softball players during the breaks, though he notes that a couple of the players are taken aback when he explains how he knows about this. Like the others in the room, he smiles and nods when Susan jumps up quickly to leave 15 minutes early.

TABLE 48.6
Two Claims Analyzed from the Problem Scenario Presented in Table 48.5

Discovering audience members' recent or ongoing technical concerns prior to organizing and delivering a software training session…

+ Enables the trainer to adapt his or her learning materials to the target audience
+ Conveys to the audience that the trainer is interested in hearing about and addressing their problems

– But the training professional may misunderstand or confuse a specific technical issue in the examples s/he develops

Browsing employees' Twitter feeds for information on their current situations…

+ Leverages well-established microblogging practices at no extra cost to the employees
+ May reveal social or personal information that enhances feelings of connection
+ May include emotional modifiers that help to prioritize issues of most concern

– But employees may be uncomfortable if they learn that their public Twitter feeds are being used for work-related purposes
– But employees' Twitter feeds may convey a variety of updates and pointers intended for a broad and diverse audience, making the work-related posts difficult to find

The scenario in Table 48.5 describes some aspects of maintaining collegial relationships across distance; it describes a typical business visit activity that can be used to communicate findings from fieldwork. Problem scenarios like this may be based directly on a particularly common or critical episode or they may be entirely synthetic. The goal is to introduce and contextualize the themes and relationships that will guide subsequent design work. This particular story combines a view of tech-savvy software developers with issues related to business travel—for example, the fact that visits are scheduled for a particular place at a particular time; there is a protocol for arriving, orienting oneself, being welcomed and integrated into a group, participants' ability to adapt their discussion in real time, and so on.

As for any scenario, the themes and relations implicit in a problem scenario can be made more explicit and open for discussion by analyzing them as claims (Table 48.6). At this point, the claims are analyzed by identifying features of the problem scenarios that have notable consequences for the actors' experiences. This is an instance of analytic evaluation and as such is clearly guided by the expectations and biases of the evaluator. A more systematic evaluation can be obtained by asking questions of the scenario that are guided by cognitive or motivational theories of human behavior (Carroll and Rosson 1992; Polson et al. 1992). The first claim explores the upsides and downsides of gathering the advance information (i.e., independent of the technology used), as this is likely to be a driving claim for the ongoing design work. Although it seems obvious that this advance preparation will be useful, it is important to acknowledge possible downsides, as they may be things that design work can remove or minimize (e.g., are there ways to make it more likely that the browser can grasp the terminology/issues he or she encounters in a post? Are there ways to increase the likelihood that the most important information will be the most obvious or easiest to find?) The second claim focuses more specifically on the discoveries made regarding Twitter use, for example, upsides and downsides of the emotional and personal content that is prevalent in Twitter and the information overload experienced by the task of browsing many feeds (Zhao and Rosson 2009).

An important characteristic of claims analysis is that it includes both positive and negative consequences. During problem analysis, there is a tendency to focus on the difficulties or concerns of current practice, for example, as observed in activity breakdowns or contradictions (Bødker 1991; Kuutti 1995; Kyng 1995; Nardi 1996). In contrast, designers tend to focus on the likely benefits of new features. Claims analysis imposes a balanced view of problems and opportunities. With respect to understanding users' requirements, taking a balanced view like this makes us aware of aspects of the current situation that are already working well. With respect to design envisionment, it forces us to consider side effects or other undesired impacts of changes to a situation.

Problem scenarios and claims are a central outcome of problem analysis in SBD. Note that these scenarios do not convey software requirements in the traditional sense of the term—they are not a specification of software features that are required. Instead, they outline a set of criteria for design requirements; whatever solutions designers propose should address the positive and negative user impacts expressed in the scenarios and claims. For instance, a microblogging solution might be "required" to reinforce the advantages of employees' existing blogging habits, but at the same time address the disadvantages of information overload. This is quite different from specifying that it will use the Twitter system. Specific features of the solution—at many levels of detail—are then identified, elaborated, evaluated, revised, or discarded in an iterative process.

48.3.2 Activity Design

Design requirements emerge and are refined throughout system development (Brooks 1995). But at some point, a team understands enough about project stakeholders and their needs that they can make specific design proposals. Indeed some projects are so over-determined that system functions are specified in advance and problem analysis consists simply of adapting these requirements to users' characteristics, work settings, and preferences. In SBD, the initial step toward specifying a design solution is made by envisioning how current activities might be enhanced or even completely transformed by available technologies. We deliberately minimize attention to the concrete steps of user interaction at this point, emphasizing the basic goals and motivations of the new activities (see also *essential use cases* as described by Constantine and Lockwood 1999).

SBD is *activity*-centered—we analyze current practice at the level of meaningful activities and build from this to new activities (Bertelsen and Bødker 2003; Kuutti and Arvonen 1992; Norman 2005). A danger in this is that the designers will focus too much on how goals are pursued in the current situation and on understanding and responding to people's current expectations about their tasks and about technology. Thus, to encourage consideration of new options and insights, we deliberately expand the "design space" when envisioning new activities. By design space, we mean the array of possible concepts and technologies that might be relevant to the problem domain, along with some analysis of what these options might bring to the design solution (MacLean, Young, and Moran 1989; Moran and Carroll 1996).

Table 48.7 exemplifies two techniques useful for exploring design alternatives. The upper part of the table shows how different conceptual metaphors evoke contrasting views of stakeholder activities. Metaphors are often used deliberately in user interface design with the hope that users will recruit them in reasoning by analogy about what a system does or how it works (Carroll and Thomas 1982; Carroll, Mack, and Kellogg 1988; Madsen 1994). Here we emphasize the role of metaphors in helping designers "think outside the box" (Erickson 1990; VerPlank 1988). Addressing real world activities and concerns is crucial to effective system design, but it is often metaphoric thinking that promotes the insights of truly creative design.

TABLE 48.7
Using Metaphors and Existing ICT Paradigms to Reason about Activities

Activity Design Features Suggested by Metaphors for Workplace Microblogging	
Hallway water cooler	Casual, easily expandable shared space; good visibility; both talk and whispers
File cabinet	Flexible system for adding, ordering, and removing items; labeled containers
Code review	Organized and issue-driven with one or more supporting artifacts; technical content
Public lecture	Content predesigned and rehearsed; multimedia; large audience; one-way channel
Activity Design Features Suggested by Information Technology	
Twitter	Content is public or by subscription; handles used for call-out to individuals; strict length limitation
Telephone	Person to person; real-time communication; utterances are transient with no archival records
E-mail listserv	List owner; hidden audience; browsable archives; posts include meta-data like sender, date, subject
File sharing	Controlled access to content items; hierarchical structure; automatic log of item uploads, downloads, edits, and so on.

An analysis of current information and communication technology (ICT) provides another approach to metaphoric thinking. When thinking about ICT, the analogy is with the classes of software and devices that already exist (e.g., web information systems, e-mail, database packages). At the same time, taking an ICT view helps to direct the design thinking back to the pragmatic concerns of software development, by enumerating possible technologies and how they might play a role in the solution. Of course, the ICT analysis will be very much influenced by starting assumptions about tools and infrastructure (see Table 48.3); for instance, if the client organization already hosts a well-established online discussion system, the designers may be expected to use that as a starting platform.

The exploration of metaphors and technology does not generate a new activity design. Rather it provides a set of lenses for discussion. The team might consider what it would be like to have a company microblog tool that shares some features with a hallway water cooler (e.g., employees can "see" when others are commenting; there is a tendency for small groups to "gather" at the same time) or with a file cabinet (e.g., posts with different content are put into different "containers"; the containers are organized in some way, perhaps using the alphabet or a hierarchy). Similarly, they may contrast the characteristics of Twitter (e.g., the tool invites personal and social content as well as work-related posts) versus e-mail (e.g., posts can be broadcast in listserv fashion, perhaps to organizational entities like a workgroup). These creative and divergent discussions form a backdrop for the convergent process of scenario envisionment.

The writing of activity scenarios is a synthetic process influenced by many factors. In some cases, one or more problem scenarios may serve as a starting point, contributing a realistic context and user goals for the overall activity. The problem scenario claims guide design reasoning; in SBD, a general heuristic is to maintain or enhance the upsides while minimizing or removing the downsides (Carroll and Rosson 1992). The metaphors and technology exploration inspire ideas and comparisons, while the claims offer constraints to be satisfied. The designers' knowledge of HCI and of interactive system design more broadly also provide important guidance; for example, knowing the relative affordances of different computer-mediated communication channels, understanding the motivational differences in discretionary versus organizationally mandated software use, and so on.

An activity design scenario for Jerry's business trip appears in Table 48.8. It reuses the hypothetical stakeholders and activity context from the problem scenario. In fact, the narrative is quite parallel to the problem scenario, but rather than envisioning the use of Twitter, it proposes a new company-hosted service. The effect of different metaphors can be seen: similar to drawers a in file cabinet, posts can be submitted with different "tags" that serve a grouping function; the idea to use a specialized "fun stuff" tag reflects the effort to promote the personal exchanges observed at a water cooler and in Twitter feeds. Other metaphors and technology are likely to affect other scenarios; for instance, we have not yet considered scenarios for the remote employees making the posts or for the managers who both make and perhaps later review the posts. Note that the scenario also addresses the claim documented in Table 48.7, in that it focuses on maintaining the benefits such as unobtrusive review and leveraging of staff microblog habits, while also addressing downsides like finding relevant content, and a way to include pointers to additional information about specialized content.

Even though activity scenarios are intentionally quite abstract, the designers can begin to evaluate them with respect to their usage implications. One way to do this is with participatory design sessions (Carroll et al. 2000; Muller 1992; Kyng 1995) that focus on how well the proposals suit stakeholders' needs. Such sessions can be structured by claims analyzed for the activity designs, asking stakeholders to weigh the pros and cons of the designs through "what if" discussions (Chin, Rosson, and Carroll 1997).

Two sample claims analyzed from the activity scenario appear in Table 48.9. Not surprisingly, they bear some resemblance to those extracted from the problem scenario; just as SBD promotes scenario elaboration and transformation, the corresponding claims often will document an evolution in the underlying design rationale (DR). For instance, the benefits

TABLE 48.8
Possible Activity Design for Jerry's Preparation and Business Visit

Activity Scenario: Jerry Visits a Remote Company Site to Carry Out a Training Session

<Background on Jerry and his upcoming visit…>

Before any training session, Jerry likes to find out what the participants have been working on lately, so that he can quickly establish common ground and use their current work as examples in his training. He decides to review everyone's posts in the new company-specific microblog tool instead of sending out a bunch of e-mails. He quickly locates the microblogs from the Seattle site and within these the three groups he will be visiting. Using the tags created by the employees he is able to grasp a few of the key issues they have been wrestling with, with plenty of evidence for the frustration they are feeling; he even finds links in some of the posts to code examples that he can incorporate into his training talk. Just as he starts to close down, he notices a "fun stuff" tag. Curious, he explores and learns that some of the staff have joined a co-ed softball team and that Susan just had a baby; he reads on to hear about her experiences at work-baby balancing and looks at baby pictures.

During his Seattle visit, the training sessions go well as he is able to introduce and discuss project-relevant examples from the start of his presentation. Like others in the room, he smiles and nods when Susan jumps up quickly to leave 15 minutes early.

TABLE 48.9
Activity Claims That Analyze Open Issues Raised by the Design Scenario

Using a set of tags to browse work-related microblogs when organizing and delivering a training session…

+ Enables the trainer to adapt his or her learning materials to the target audience
+ Conveys to the audience that the trainer is interested in hearing about and addressing their problems
+ Offers a summary view to the reader via the tag name and grouping structure
− But the training professional may misunderstand or confuse the meaning of a tag
− But tag creation or selection may be too costly for microblog authors to adopt

Browsing employees' company-hosted microblogs for information on their current situations…

+ Leverages well-established microblogging practices
+ Increases the likelihood that work-relevant feeds will be discovered
− But the specialized tool adds an extra task for employees to remember and enact
− But these private feeds may focus entirely on work updates, and meaningful social or personal updates may be missed

of unobtrusive preparation for Jerry are still in place, but now some of the downsides noted for weeding through many non-work tweets have been addressed. In place of that downside, we have noted the likelihood that personal posts will be less likely, even though the scenario envisions a "fun stuff" tag to encourage such content. We also note that by introducing a new company-specific social media tool such as this, the company is implicitly introducing a new "reporting" task for employees to consider and manage on an ongoing basis. These downsides play an important role, as they raise issues that should be discussed and either accepted as necessary costs or addressed through additional design ideas as the project continues.

It is important to note how much progress can be made even at this very early level of envisioning activities. The narrative in Table 48.8 is concrete and evocative; designers or their clients can readily understand what is being proposed and begin to consider the relevant pros and cons. Although the influence of the metaphors and information technology can be seen, few details are provided about how posts and tags will be created or presented or how employees will access and interact with this information. At this point, the scenario is "just talk"; indeed if they are shared and discussed over an interactive medium, it might quickly be extended or revised as part of a real-time design review and discussion.

48.3.3 User Interaction Design

As design continues, tentative decisions are made about the design direction, but now starting to explore the options and tradeoffs associated with the details of the user interaction. As for activity design, we begin by exploring useful metaphors and technology. The metaphors applied to user interaction design often overlap with those from activity design, but the emphasis shifts to how users will see, understand, and respond to the system; the concerns of this phase are similar to those in the "gulfs of evaluation and execution" discussed by Norman (1986). For instance, the three metaphors of water cooler, file cabinet, and code review suggest these design ideas:

- *Water cooler*: posts are more available to people who are "there," perhaps because they appear in larger fonts or arrive with an auditory alert; some posts may be unavailable even to those currently online, as in a whisper perhaps there is a mechanism for a temporary "personal/private" tag; individuals can easily arrive or leave the shared posting space but their coming and going is indicated through some sort of alert (e.g., akin to footsteps).
- *File cabinet*: posts that fall into the same group appear under a label that can be seen when adding or reading new content; authors may be allowed to "grab" a post and move it to another labeled group.
- *Code review*: once a sufficient number of colleagues are "listening," a particular person can "take the floor," with the result that attention is directed to his/her posts in lieu of making other comments; this series of posts may refer to one or more secondary artifact that are being "presented"; other attendees can reply or question a specific post or group of posts.

Technology options explored at this phase might include hyperlinks (icons or other controls used to connect to other posts or external documents); auditory cues (earcons that signal the initiation of a "meeting," or the arrival/departure of colleagues as they activate the microblog tool); and a variety of user interaction control widgets (e.g., a gauge used to set a "whisper" level for a post).

Ideas such as these are discussed and synthesized under the general guidance user interaction design experience and guidelines. Information and interaction design possibilities may be "tried out" in the context of various activity design scenarios, with attention directed at claims analyzed in earlier phases as well as new features and consequences. Does the design reduce the cost of creating or browsing tag structures? Does someone who chooses to post about an issue in more depth feel s/he has an audience? Do the privacy controls encourage more informal banter and exchange? Ongoing and creative design inquiry like this—and the scenarios that provide a real world context for the reflective process—is a hallmark of SBD.

One vision of user interaction for the activity design appears in Table 48.10. For simplicity, we elaborate just the tag browsing subtask. The scenario offers a view into a company microblog that includes a hierarchical information display (org chart) as well as a network display (tag cloud) that can be used to access relevant content. It assumes that posts are browsed as normal in a sequential fashion. It also envisions some specific user interaction details that support sensemaking and user action, for instance an overview of the group's blogging activity positioned in a salient "starting" position and the use of font size and shading to convey popularity and recency of tags.

In parallel with the exploration of these or other user interaction details, the design team would use claims analysis to document the pros and cons raised by the more specific proposals. Although we will not elaborate this example any further in the interest of simplicity, it is clear what some of these tradeoffs might be: a header provides an overview but takes space away from the main content, which is the posts; a tag cloud that pops up over the main display can be a visual distraction and adds window management as a subtask; seeing all the tags at once may be overwhelming (e.g., if there are many) and lead to feelings of information overload. These sorts of issues would be tracked and discussed as the scenario elaboration process continues.

Ultimately, user interaction design comprises all aspects of how the activities' objects and actions are rendered and executed during users' activities. In many design projects, this may include special attention to the needs of new or inexperienced users. For instance, suppose that this was Jerry's first use of a microblogging system. How would he know to

TABLE 48.10
Piece of a User Interaction Scenario for Jerry's Use of the Microblogging Tool

User Interaction Scenario: Jerry Visits a Remote Company Site to Carry Out a Training Session

<Background on Jerry and his upcoming visit, goals and decision to review the posts…>

Jerry starts up the company microblog tool, which by default opens to his own workgroup. He calls up an org chart, pans the map to locate the Seattle lab, then finds Susan's group (her name and group title appear when he hovers over her node with his mouse). When he double-clicks to open her group, all posts from her staff appear, with most recent posts at the top of the list. In the header, he can see summary information, including the number of blogs in the past week, the number of staff who have contributed, and the top five tags that have been active over that time.

Jerry knows he can access subsets of tags by selecting any of the top five but instead decides to first get an overview by asking for a tag cloud. This causes a secondary display to pop up, showing all tags used by this group. The size of the tag shows its overall use and its shading (more or less saturated) shows how recently it has been used. He can quickly see that several tags related to graphics transformations and data consistency are a recent popular topic, so he decides to begin with these feeds. He can open each subset by double-clicking its tag in the cloud. This helps him to grasp the key issues they have been wrestling with, with plenty of evidence for the frustration they are feeling; he even finds links to code examples he can incorporate into his training talk. Before closing the group, he takes a bit of time to find and browse a few more personal blog sets, including the "fun stuff" and "family news" tags.

<Ending that conveys the benefits of the microblog browsing Jerry has done, including helpfulness of the personal information>

get an overview via a tag cloud? The header view might need some special indication, perhaps as simple as a "More…" link next to the top five tags, to invite such a goal. It can be useful to create "user help" versions of design scenarios, where an actor(s) is assumed to have little or no experience using the system. In SBD, help prompts and similar guidance for new users may also be inspired by metaphors (e.g., a coach) and technology (e.g., online tutorials, animated demonstrations) (Rosson and Carroll 2002; Chapter 8).

48.3.4 Evaluation-Centered Iterative Design

Like most user-centered design frameworks, SBD assumes that usability evaluation begins early and continues throughout design and development and that a variety of design representations will be evaluated. For instance, the process of creating user interaction design scenarios often includes the creation of user interaction prototypes; these in turn can be used for quite formal usability testing (e.g., representative users carry out representative tasks on early operational prototypes). Formal evaluations require sufficient progress on a design to enable construction of a prototype, though of course user interaction prototypes can be created in special-purpose languages or tools or may be low-fidelity prototypes constructed of cardboard and paper (Muller 1991; Virzi, Sokolov, and Karis 1996).

Starting in the very early phases of design, user feedback may be obtained in informal participatory design sessions (Muller 1992). It is quite possible to include users in discussion and analysis of problem scenarios and in initial envisionment of activity scenarios (Chin, Rosson, and Carroll 1997). The design ideas are also subjected constantly to analytic evaluation through claims analysis and other design review activities (e.g., usability inspections or cognitive walkthrough; Nielsen 1995; Nielsen and Mack 1994; Polson et al. 1992). All these activities yield formative evaluation feedback that guides changes and expansion of the design vision.

SBD builds on Scriven's (1967) concept of *mediated evaluation*. In mediated evaluation, empirical data are collected (Scriven calls this "pay-off" evaluation), but the materials and methods of the empirical tests are guided by prior analytic evaluation. The analytic evaluation may have many different components, for example, an expert-based inspection or perhaps a cognitive model constructed for a particularly complex or critical interaction sequence (Gray, John, and Atwood 1992; Kieras 1997). In SBD, claims analysis is used to mediate empirical evaluations. Claims written during scenario generation and discussion document the usability issues thought to be most likely to influence users' success, failure, and satisfaction; thus, the claims are used as a skeleton for constructing and administering user tests. In fact one view of a claims set is as a series of usability hypotheses that can be assessed empirically; claims also suggest *why* a design feature may have a particular impact on users' experiences.

Rosson and Carroll (2002) describe how to use scenarios and claims in a systematic way to generate *usability specifications*, a set of baseline tasks with performance and user satisfaction targets that can be assessed in an iterative fashion as the system design evolves and stabilizes. Usability specifications like this have two important roles in evaluation. First, they provide concrete usability-centered objectives that serve as a management tool in system development—if a product team accepts the target measures, then a team's usability engineers are able to insist that redesign and improvement continue until they are met (Carroll and Rosson 1985; Good et al. 1986).

Second, the specifications tie the results of empirical evaluation directly to the usability issues raised during design. For instance, the user interaction scenario specified that Jerry grasps an overview of topics under discussion by Susan's team by looking at the tag cloud. This claim might be extracted as a benchmark task and used to evaluate the efficacy of various tag visualization designs, with the goal of optimizing the comprehension impacts for the user.

48.4 APPLICATIONS FOR SCENARIO-BASED DESIGN

As exemplified by books discussing scenario-based methods (e.g., Alexander and Maiden 2004; Carroll 1995, 2000), one strength of design scenarios is the many roles they can play in system development. Scenarios are accessible to many process stakeholders and as such can be a tremendous aid to project communication and coordination. Scenarios can be represented at varying levels of specificity and formality, making them an attractive medium for gradual evolution and specification of a system's functions. They can also capture important aspects of the nonfunctional constraints or requirements for a project. In this section, we briefly survey research and practice that is exploring different applications of scenarios to the problems of software development.

48.4.1 Scenarios in the System Development Lifecycle

Our example has focused on the familiar processes of requirements analysis, design, and usability evaluation, but scenario-based methods support many diverse goals in system development (Carroll 1997; Carroll et al. 1998). Product planners present day-in-the-life scenarios to managers as design visions (Dubberly and Mitsch 1992); requirements engineers gather workplace scenarios through direct observation and interviews and analyze scenarios as primary data (Antón, McCracken, and Potts 1994; Holbrook 1990; Hsia et al. 1994; Kaindl 1997; Kuutti 1995; Potts 1995).

Even if scenarios are not developed and incrementally refined as recommended in the SBD framework, they may be used at many points along the way. For instance, task-based user documentation is often structured by scenarios. Minimalist help and training provide many examples of this, such as a "training wheels" system that blocks functions that are not relevant to a paradigmatic novice-use scenario (Carroll and Carrithers 1983) or a "view matcher" that guides new programmers through a predefined scenario of debugging and modification (Carroll et al. 1990; Carroll and Rosson 1991; Rosson and Carroll 1996).

48.4.1.1 Personas and Envisionment

The concept of a persona—the envisionment and elaboration of a hypothetical target user with her own personality, needs, and preferences—was originally popularized by Alan Cooper and his colleagues (Cooper 1998; Cooper and Reimann 2003). Personas are often developed during the problem analysis or requirements specification phase of a project, as a means for understanding, expressing, and working with the goals and implied requirements of different target users. Prospective users are conveyed through detailed composite user archetypes (often represented visually, including personality characteristics) and with a context scenario that helps to anticipate user needs and expectations, so that the designers working with the personas can have as direct an experience as possible with potential users. Although it is not yet clear exactly what elements of a persona are most important in guiding or inspiring design, research does suggest that software designers find personas to be of use in the early phase of design when they are trying to understand users' needs (Vasara 2003).

An important issue discussed by Turner and Turner (2011) is the impact of stereotypes on designing with personas—that is, to the extent that personas are "concocted" from a variety of diffuse information, will the design team's own biases or wishful thinking have too great an influence? They argue that while stereotypes are often surprisingly accurate, the creation of personas without sound empirical backing should be approached as a sort of design dialectic, where contrasting frames (perhaps even using extreme or marginal stakeholders) are swapped in and out of the reasoning space to keep it lively and provocative. This recommendation is analogous to the use of "what if?" reasoning in SBD to drive the analysis of tradeoffs, in the hopes of expanding the design space beyond what is already familiar to the designers.

Clearly persona-centered design are similar in many ways to SBD, although we prefer to use the phrase "hypothetical stakeholder," emphasizing that these characters are imaginary and that they may include a diverse set of perspectives and roles in the problem situation under consideration. But like personas, SBD encourages designers to become familiar with and empathize with envisioned characters and personal attributes who are used repeatedly in the iterative process of scenario analysis, envisionment, and refinement.

48.4.1.2 Use Cases and Software Engineering

Scenarios have also come to play a central role in object-oriented software engineering (Alexander and Maiden 2004; Jacobson 1995; Jacobson, Booch, and Rumbaugh 1998; Rubin and Goldberg 1992; Wirfs-Brock, Wilkerson, and Wiener 1990). A *use case* is a scenario written from a functional point of view, enumerating all the possible user actions and system reactions that are required to meet a proposed system function (Jacobson et al. 1992). Use cases can then be analyzed with respect to their requirements for system objects and interrelationships. Wirfs-Brock (1995) describes a variant of use case analysis in which she develops a "user-system conversation": using a two-column format, a scenario is decomposed into a linear sequence of inputs from the user and the corresponding processing and/or output generated by the system. Kaindl (2000) extends this analysis by annotating how scenario steps implement required user goals or system functions.

Scenarios are promising as a mediating representation for analyzing interactions between human-centered and software-centered object-oriented design issues (Rosson and Carroll 1993, 1995, 2001). As we have seen, scenarios can be decomposed with respect to the software objects needed to support the narrated user interaction. These software objects can then be further analyzed with respect to their system responsibilities, identifying the information or services that should be contributed by each computational entity (Wirfs-Brock and Wilkerson 1989; Beck and Cunningham 1989; Rosson and Gold 1989). This analysis (often termed *responsibility-driven design*; Wirfs-Brock, Wilkerson, and Wiener 1990) may lead to new ideas about system functionality, for example, initiatives or actions taken by a software object on behalf of the user or another object. Scenarios and claims analysis are useful in describing these new ideas and considering their usability implications *in the context of use* (Rosson 1999; Rosson and Carroll 1995).

Beyond the natural relationship of scenarios to use cases, software engineers working to refine methods for agile development have found that scenarios can play an important role. Of course, agile methods have always included an important role for user stories or other narratives that serve as a flexible and evolving source of user-centered requirements. However, Lee, McCrickard, and Stevens (2009) argue for expanding this general role through more systematic use of scenarios and claims analysis; they propose the more rigorous agile method of eXtreme SBD (XSBD) and demonstrated its usefulness in a case study of a commercial product.

48.4.1.3 Functional and Nonfunctional Requirements

The general accessibility of scenarios and claims makes them an excellent medium for raising and discussing a variety of competing concerns. We have emphasized their role in conceptualizing opportunities and challenges facing the design team and in negotiating among competing options as new ideas emerge and are evaluated. Others have used these methods in a variety of domains, including children's digital libraries (Theng et al. 2002); negotiation support systems (Pommeranz et al. 2009); personal health management (Rogers 2009); and driver support systems (Tideman, van der Voort, and van Arem 2009). Swanson, Sato, and Gregory (2009) show that SBD can be adapted to discover and address the influences of different and partially overlapping cultural contexts in interactive system design. More generally, both McCall (2010) and Sutcliffe (2010) argue that claims analysis (i.e., considering pros and cons for any feature under consideration) can serve as a mechanism to promote creativity in design reasoning.

Beyond their focal concern for specifying appropriate and useful functionality for a system under design, software engineers are concerned about issues such as code reuse, programming language or platform, and so on; management is concerned with project resources, scheduling, and so on; a marketing team focuses on issues such as the existing customer base and the product cost. These diverse concerns are *nonfunctional requirements* on system development—concerns about "how" a system will be developed, fielded, and maintained rather than "what" a system will provide (Sommerville 1992). Usability goals are often specified as nonfunctional requirements, in that they typically focus on the quality of the system rather than its core functions (Mylopoulos, Chung, and Nixon 1992). The low cost of development, content flexibility, and natural language format of scenarios and claims make them excellent candidates for contrasting and discussing a range of such issues throughout the software development lifecycle (Sutcliffe and Minocha 1998). A particularly interesting case study is presented by Maiden and his colleagues (Maiden et al. 2007), wherein scenarios and associated walkthroughs were useful in discovering airport requirements that could minimize environmental impact.

Taking a more formal approach, a scenario can be modeled as a sequence of task steps and this sequence can be analyzed with respect to users' ability to perform the task under different circumstances (Alexander and Maiden 2004). For example, Gregoriades and Sutcliffe (2005) illustrate the use of Bayesian Belief Network models for assessing the reliability of human users to carry out a scenario's steps under varying environmental pressures. By noting when and where the predicted reliability fails to pass a benchmark threshold, the analysts can identify problematic aspects of a proposed scenario and contrast scenarios from alternative designs. The goals of this approach overlaps with tradeoff analysis, but it provides a more systematic and mathematical representation of design issues most likely to need attention in a design.

48.4.1.4 Evaluating Systems in Use

In SBD, a software development process begins with fieldwork that ultimately is summarized through a set of stakeholder profiles and diagrams, themes, scenarios, and claims analysis (Rosson and Carroll 2002; Chapter 2). However, researchers are currently exploring a variant of this approach in which scenarios are used as an elicitation technique for capturing current practices and reactions during the adoption and acceptance phase of a fielded system. Haynes, Purao and Skattebo (2004, 2009) report a field study of a collaborative system for product lifecycle management in the U.S. Marine Corps. They interviewed 26 current users of the system, selecting individuals representing the various organizational roles associated with system use (e.g., engineers, scientists, staff members). Their interview script was scenario-based: after eliciting information about the interviewee's work context, they asked for personal scenarios of system use. Later they transcribed the interviews and coded the scenarios to identify system features and users' apparent concerns about these features (coded as claims with upsides and downsides). The findings were summarized and shared through focus groups to refine the evaluation. The technique allowed them to collect and summarize a number of related concerns, and also to connect (and on occasion illustrate) these concerns with the organizational roles and usage context represented by the user scenarios.

As this example suggests, it is quite possible to leverage the expressive and communicative power of scenarios in various ways, even when a user-centered analyst enters a software development process that is well underway. Because scenarios are focused on the specifics of a usage context and associated user experience, they are a natural medium for "war stories" about how things are or are not working (Orr 1986). In a recent application of scenario-based evaluation, Haynes and his colleagues (Haynes, Spence, and Lenze 2009) showed that university students can write two sorts of learning scenarios at the end of a course—describing both valuable learning experiences and experiences that should be improved. The educators found that the scenario-writing was useful as a reflective activity for the students but also that the narratives were rich representations for reasoning about course improvements or redesign. When complemented by tradeoff analysis (e.g., documented as claims), such scenarios can become a powerful medium for discussing the need for change, whether aimed at the nature of people's goals and activities or to the technology that supports them.

Other researchers have been contrasting the effectiveness of claims analysis as an analytic evaluation method with other task-centered usability methods. For instance, Blandford and Artfield (2010) summarize a series of studies that compared claims analysis against the popular methods of heuristic evaluation and cognitive walkthrough. They found that claims analysis required more skill on the part of the evaluator (i.e., to recognize and document a rich set of tradeoffs for critical features of a design), it revealed issues at a deeper and more conceptual level of analysis than heuristic evaluation or cognitive walkthrough. They also stress the need to embed any evaluation task in scenarios of use that highlight "those aspects of context that will result in ecologically valid task performance" (p. 54). The work of Theng and colleagues (Theng et al. 2002) is consistent with such claims; they showed that SBD with claims analysis could be used in an intergenerational participatory design setting and were effective at directing attention to important theory-based issues.

48.4.2 Scenario-Based Design Rationale and Reuse

To this point, we have focused on the instrumental role of scenarios and tradeoff analyses in supporting the analysis, design, and evaluation activities of interactive software development. However, another potential role for SBD is building and expressing a science base for user-centered design. In this role, a system's scenario-based analyses comprise DR that can be recruited for explanation, generalization, and reuse—contributing to the theory base for user interaction design (Sutcliffe 2002). Software engineers have often used design patterns as containers of such knowledge for reuse, and recent work by Abraham and Atwood (2009) suggests that a combination of patterns with claims may be particularly effective in recognizing and working with design tradeoffs.

Carroll and Rosson (2003) use a community MOO as a design case study that illustrates the three scientific foundations of scenario-based DR: action science, ecological science, and synthetic science. The MOO's scenarios and associated claims supported action science during the envisioning process, when novel usage features were imagined in response to the real world needs and preferences of users. The DR supported ecological science by surveying and documenting salient causal features of real world usage contexts; these surveys can guide design efforts in similar activities or domains in the future. Finally, the contributions to synthetic science came through the association of interdisciplinary evidence and explanations to the claims' upsides and downsides.

A more complete analysis of scenarios and claims as theoretical material can be found in Sutcliffe's volume on Domain Theory (2002). He argues that claims in particular are an ideal abstraction for stating reusable knowledge about interaction design that is grounded in the relevant usage context. He proposes a template for developing and organizing claim-based knowledge and argues through demonstration that such a representation is reusable from one design project to another. This potential for claims as reusable design knowledge has been explored more thoroughly by researchers developing the LINK-UP claims library and scenario-based development tool (Chewar et al. 2005; Payne et al. 2003). These researchers have elaborated the claim format proposed by Sutcliffe and have been using it to collect and organize scientific knowledge about notification system design. They are currently exploring the effectiveness of the tools in teaching usability engineering concepts and skills to undergraduate and graduate students (Chewar and McCrickard 2003).

48.4.3 Teaching Usability with Scenario-Based Design

Another indirect benefit of scenario-based development is that the resulting scenarios, claims, and related design documents can support a variety of case-based learning activities. As a concrete story, even a single scenario can be evocative enough to illustrate and discuss the usage issues raised by its tradeoff analysis. A set of interrelated and successively refined scenarios and tradeoffs presents an open-ended and rich information structure for a variety of active learning experiences (Carroll and Rosson 2005a,b,c).

Over the past few years, we have been collaborating with other HCI faculty and practitioners to build a case library of scenario-based usability engineering projects (ucs.ist.psu.edu) and incorporating case-based learning activities into our graduate and undergraduate teaching. The students carry out a number of in-class or homework assignments that involve case analysis over the course of a semester—for example, analyzing the implications of "perturbing" a case study in specific ways or tracing the impacts of a tradeoff analyzed early in development through the lifecycle. They also analyze, design, and prototype their own interactive projects, documenting their work as a miniature case study. The case-based experiences serve as a sort of surrogate for apprenticeship learning in the real world, as the students are able to explore and question the reasoning and decisions of actual development teams. At the same time, the cases are an excellent vehicle for encountering and weighing the many competing concerns that development teams must address during interactive system development.

48.5 CURRENT CHALLENGES

When we design interactive systems, we make use. We create possibilities for learning, work, and leisure, for interaction and information. Scenarios—descriptions of meaningful usage episodes—are popular representational tools for making use. They help designers to understand and to create computer systems and applications as artifacts of human activity, as things to learn from, as tools to use in one's work, as media for interacting with other people.

SBD offers significant and unique leverage on some of the most characteristic and vexing challenges of design work: Scenarios are at once concrete and flexible, helping developers manage the fluidity of design situations. Scenarios emphasize the context of work in the real world; this ensures that design ideas are constantly evaluated in the context of real world activities, minimizing the risk of introducing features that satisfy other external constraints. The work-oriented character of scenarios also promotes work-oriented communication among stakeholders, helping to make design activities more accessible to many sources of expertise. Finally, scenarios are evocative, raising questions at many levels, not only about the needs of the people in a scenario as written, but also about variants illustrating design tradeoffs.

Scenario-based methods are not a panacea. A project team who complains "We wrote scenarios, but our system still stinks!" must also report how their scenarios were developed, who reviewed them, and what roles they played in system development. If a user interaction scenario is not grounded in what is known about human cognition, social behavior, and work practices, it may well be inspiring and evocative, but it may mislead the team into building the wrong system (Carroll et al. 1998). Scenarios are not a substitute for hard work.

At the same time, *any* work on user interaction scenarios directs a project team to the needs and concerns of the people who will use a system. It is in this sense that scenarios can provide a very lightweight approach to human-centered design. Simply writing down and discussing a few key expectations about users' goals and experiences will enhance a shared vision of the problems and opportunities facing system users. Adopting a more systematic framework such as described here adds control and organization to the creative process of design and at the same time generates work products (scenarios and claims) that can serve as enduring DR during system maintenance and evolution (McKerlie and MacLean 1994; Moran and Carroll 1996).

Where are scenarios taking us? The current state of the art in the design of interactive systems is fragmented. Scenarios are used for particular purposes throughout system development, but there is no comprehensive process (Carroll 1995; Jarke, Bui, and Carroll 1998; Weidenhaupt et al. 1998). Scenario practices have emerged piecemeal, as local innovations, leading to a considerable variety of scenario types specialized for particular purposes (Campbell 1992; Young and Barnard 1987). A detailed textual narrative of observed workplace practices and interactions, a use case analysis of an object-oriented domain model, a day-in-the-life video envisionment of a future product, and the instructions for test subjects in an evaluation experiment could *all* be scenarios. Recognizing this, and cross-leveraging the many different views of scenarios, is a potential strength of SBD. But much work remains in developing overarching frameworks and methods that exploit this potential advantage.

It is important for us to be ambitious, skeptical, and analytic about scenarios and SBD. Almost 50 years ago, Herman Kahn (1962) expressed puzzlement that scenarios were not more widely used in strategic planning. As we write this in 2010, scenarios have become so pervasive in interactive system design that younger designers may wonder what the alternative is to SBD! But there is yet some strangeness to scenarios. We are not much farther than Kahn was in understanding how scenarios work as tools for planning and design or in understanding how to fully exploit their unique strengths as aides to thought.

REFERENCES

Abraham, G., and M. E. Atwood. 2009. Patterns or claims: Do they help in communicating design advice?. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7 (Melbourne, Australia, November 23–27, 2009). OZCHI '09*, Vol. 411, 25–32. New York: ACM. DOI= http://doi.acm.org/10.1145/1738826.1738831.

Ackoff, R. L. 1979. Resurrecting the future of operations research, *J Oper Res Soc* 30(3):189–99.

Alexander, I. F., and N. Maiden. 2004. *Scenarios, Stories, Use Cases through the Systems Development Life-Cycle*. London: Wiley.

Antón, A., W. M. McCracken, and C. Potts. 1994. Goal decomposition and scenario analysis in business process reengineering. In *Proceedings of CAiSE'94: Sixth Conference on Advanced Information Systems Engineering*. Utrecht, The Netherlands: Springer-Verlag.

Beck, K. 1999. *Extreme Programming Explained: Embrace Change*. Reading, MA: Addison-Wesley.

Beck, K., and W. Cunningham. 1989. A laboratory for teaching object-oriented thinking. In *Proceedings of Object-Oriented Systems, Languages and Applications: OOPSLA '89*, ed. N. Meyrowitz, 1–6. New York: ACM.

Bertelsen, O. W., and S. Bødker. 2003. Activity theory. In *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science*, ed. J. M. Carroll, 291–324. San Francisco, CA: Morgan Kaufmann.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: A Customer-Centered Approach to System Design*. San Francisco, CA: Morgan Kaufmann.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: Defining Customer-Centered Systems*. San Francisco, CA: Morgan Kaufmann.

Blandford, A., and S. Attfield. 2010. *Interacting with Information*. San Francisco, CA: Morgan and Claypool Publishers.

Bødker, S. 1991. *Through the Interface: A Human Activity Approach to User Interface Design*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Brooks, F. 1995. *The Mythical Man-Month: Essays on Software Engineering*. Reading, MA: Addison-Wesley. (Anniversary Edition, originally 1975).

Campbell, R. L. 1992. Will the real scenario please stand up?. *SIGCHI Bull* 24(2):6–8.

Carroll, J. M. eds. 1995. *Scenario-Based Design: Envisioning Work and Technology in System Development*. New York: John Wiley and Sons.

Carroll, J. M. 1997. Scenario-based design. In *Handbook of Human-Computer Interaction*, Second ed., ed. M. Helander and T. K. Landauer, 383–406. Amsterdam: North Holland.

Carroll, J. M. 2000. *Making Use: Scenario-Based Design of Human-Computer Interactions*. Cambridge, Massachusetts: MIT Press.

Carroll, J. M., and C. Carrithers. 1983. Blocking errors in a learning environment. In *Psychonomic Society 24th Annual Meeting (San Diego, CA, November 17)*, Abstract in Proceedings, 356. New York: Springer.

Carroll, J. M., G. Chin, M. B. Rosson, and D. C. Neale. 2000. The development of cooperation: Five years of participatory design in the Virtual School. In *Proceedings of DIS 2000: Designing Interactive Systems*, 239–51. Brooklyn, NY: ACM.

Carroll, J. M., J. Karat, S. A. Alpert, M. van Deusen, and M. B. Rosson. 1994. Demonstrating raison d'etre: Multimedia design history and rationale. In *CHI'94 Conference Companion*, ed. C. Plaisant, 29–30. New York: ACM.

Carroll, J. M., R. L. Mack, and W. A. Kellogg. 1988. Interface metaphors and user interface design. In *Handbook of Human-Computer Interaction*, ed. M. Helander, 67–85. Amsterdam: North Holland.

Carroll, J. M., and M. B. Rosson. 1985. Usability specifications as a tool in iterative development. In *Advances in Human-Computer Interaction*, ed. H. R. Hartson, 1–28. Norwood, NJ: Ablex.

Carroll, J. M., and M. B. Rosson. 1990. Human-computer interaction scenarios as a design representation. In *Proceedings of the 23rd Annual Hawaii International Conference on Systems Sciences (Kailua-Kona, HI, January 2–5)*, 555–61. Los Alamitos, CA: IEEE Computer society Press.

Carroll, J. M., and M. B. Rosson. 1991. Deliberated evolution: Stalking the view matcher in design space. *Hum Comput Interact* 6:281–318.

Carroll, J. M., and M. B. Rosson. 1992. Getting around the task-artifact cycle: How to make claims and design by scenario. *ACM Trans Inf Syst* 10:181–212.

Carroll, J. M., and M. B. Rosson. 2003. Design rationale as theory. In *HCI Models, Theories, and Frameworks: Toward an Interdisciplinary Science*, ed. J. M. Carroll, 431–61. San Francisco, CA: Morgan Kaufmann.

Carroll, J. M., and M. B. Rosson. 2005a. Case studies as minimalist information designs. In *Proceedings of HICSS 38: Hawaii International Conference on Systems Science*, Hilton Waikoloa Village: IEEE Digital Library.

Carroll, J. M., and M. B. Rosson. 2005b. A case library for usability engineering: Development, experiences, and outcomes. *ACM J Educ Res Comput* 5(1): Article 3, 1–22.

Carroll, J. M., and M. B. Rosson. 2005c. Toward even more authentic case-based learning. *Educ Technol* 45(6):5–11.

Carroll, J. M., M. B. Rosson, G. Chin, and J. Koenemann. 1998. Requirements development in scenario-based design. *IEEE Trans Softw Eng* 24(12):1156–70.

Carroll, J. M., J. A. Singer, R. K. E. Bellamy, and S. R. Alpert. 1990. A view matcher for learning smalltalk. In *Proceedings of CHI90: Human Factors in Computing Systems (Seattle, WA, April 1–5)*, 431–7. New York: ACM.

Carroll, J. M., and J. C. Thomas. 1982. Metaphors and the cognitive representation of computing systems. *IEEE Trans Syst Man Cybern* 12(2):107–16.

Checkland, P. B. 1981. *Systems Thinking, Systems Practice*. Chichester: John Wiley.

Chewar, C. M., E. Bachetti, D. S. McCrickard, and J. Booker. 2005. Automating a design reuse facility with critical parameters: Lessons learned in developing the LINK-UP system. In *Computer-Aided Design of User Interfaces IV*, ed. R. Jacob, Q. Limbourg, and J. Vanderdonckt, 235–46. Norwell, MA: Kluwer Academic Publishers.

Chewar, C. M., and D. S. McCrickard. 2003. Educating novice developers of notification systems: Targeting user-goals with a conceptual framework. In *Proceedings of ED-MEDIA '03*, 2759–66. Chesapeake, VA: AACE.

Chin, G., M. B. Rosson, and J. M. Carroll. 1997. Participatory analysis: Shared development of requirements from scenarios. In *Proceedings of Human Factors in Computing Systems, CHI'97 Conference*, 162–9. New York: ACM.

Constantine, L. L., and L. A. D. Lockwood. 1999. *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*. Reading, Massachusetts: Addison-Wesley.

Cooper, A. 1999. *The Inmates are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. Indianapolis, IN: SAMS Press.

Cooper, A., and R. Reimann. 2003. *About Face 2.0*. Indianapolis, IN: Wiley Publishing.

Cross, N. 2001. Design cognition: Results from protocol and other empirical studies of design activity. In *Design Knowing and Learning: Cognition in Design Education*, ed. C. Eastman, M. McCracken and W. Newstetter, 79–103. Amsterdam: Elsevier.

Dubberly, H., and D. Mitsch. 1992. Knowledge navigator. In *CHI'92 Special Video Program: Conference on Human Factors in Computing Systems (Monterey, CA, May 3–7)*, ed. B. A. Myers, New York: ACM SIGCHI.

Erickson, T. 1990. Working with interface metaphors. In *The Art of Human-Computer Interface Design*, ed. B. Laurel, 65–73. Reading, MA: Addison-Wesley.

Erickson, T. 1995. Notes on design practice: Stories and prototypes as catalysts for communication. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 37–58. New York: John Wiley and Sons.

Freud, S. 1900. *The Interpretation of Dreams, Standard Edition, Vol. IV*, Hogarth, London.

Good, M., T. M. Spine, J. Whiteside, and P. George. 1986. User-derived impact analysis as a tool for usability engineering. In *Proceedings of Human Factors in Computing Systems: CHI '86*, ed. M. Mantei and P. Oberton, 241–6. New York: ACM.

Gray, W. D., B. E. John, and M. E. Atwood. 1992. The precis of Project Ernestine, or an overview of a validation of GOMS. In *Proceedings of Human Factors in Computing Systems: CHI '92*, ed. P. Bauersfeld, J. Bennett, and G. Lynch, 307–12. New York: ACM.

Gregoriades, A., and A. Sutcliffe. 2005. Scenario-based assessment of nonfunctional requirements. *IEEE Transact Softw Eng* 31(5):392–409.

Haviland, S. E., and H. H. Clark. 1974. What's new? Acquiring new information as a process in comprehension. *J Verbal Learn Verbal Behav* 13:512–21.

Haynes, S. R., S. Purao, and A. L. Skattebo. 2004. Situating evaluation in scenarios of use. In *Proceedings of CSCW 2004*, 92–101. New York: ACM.

Haynes, S. R., S. Purao, and A. L. Skattebo. 2009. Scenario-based methods for evaluating collaborative systems. *Comput Support Coop Work* 18(4):331–56. DOI= http://dx.doi.org/10.1007/s10606-009-9095-x.

Haynes, S. R., L. Spence, and L. Lenze. 2009. Scenario-based assessment of learning experiences. In *Proceedings of the 39th IEEE international Conference on Frontiers in Education Conference (San Antonio, Texas, USA, October 18–21, 2009)*. 870–6. Piscataway, NJ: IEEE Press.

Holbrook, H. 1990. A scenario-based methodology for conducting requirements elicitation. *ACM SIGSOFT Softw Eng Notes* 15(1):95–103.

Hsia, P., J. Samuel, J. Gao, D. Kung, Y. Toyoshima, and C. Chen. 1994. Formal approach to scenario analysis. *IEEE Softw* 11(2):33–41.

Jacobson, I. 1995. The use-case construct in object-oriented software engineering. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 309–36. New York: John Wiley & Sons.

Jacobson, I., G. Booch, and J. Rumbaugh. 1998. *The Unified Software Development Process*. Reading, MA: Addison-Wesley.

Jacobson, I., M. Christersson, P. Jonsson, and G. Övergaard. 1992. *Object-Oriented Software Engineering—A Use-Case Driven Approach*. Reading, MA: Addison-Wesley.

Jarke, M., X. T. Bui, and J. M. Carroll. 1998. Scenario management: An interdisciplinary approach. *Requir Eng* 3(3–4):155–73.

Kahn, H. 1962. *Thinking about the Unthinkable*. New York: Horizon Press.

Kahneman, D., and A. Tversky. 1972. Subjective probability: A judgement of representativeness. *Cogn Psychol* 3:430–54.

Kaindl, H. 1997. A practical approach to combining requirements definition and object-oriented analysis. *Ann Softw Eng* 3:319–43.

Kaindl, H. 2000. A design process based on a model combining scenarios with goals and functions. *IEEE Trans Syst Man Cybern* 30(5):537–51.

Karat, J. 1995. Scenario use in the design of a speech recognition system. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 109–33. New York: John Wiley & Sons.

Karat, J., and J. B. Bennett. 1991. Using scenarios in design meetings—a case study example. In *Taking Design Seriously: Practical Techniques for Human-Computer Interaction Design*, ed. J. Karat, 63–94. Boston, MA: Academic Press.

Kieras, D. 1997. A guide to GOMS model usability evaluation using NGOMSL. In *Handbook of Human-Computer Interaction*, Second ed., ed. M. G. Helander, T. K. Landauer, and P. V. Pradhu, 733–66. Amsterdam: North-Holland.

Kraut, R., R. Fish, R. Root, and B. Chalfonte. 1993. Informal communication in organizations: Form, function, and technology. In *Proceedings of CSCW'93: Conference on Computer-Supported Cooperative Work*. New York: ACM.

Kuutti, K. 1995. Work processes: Scenarios as a preliminary vocabulary. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 19–36. New York: John Wiley & Sons.

Kuutti, K., and T. Arvonen. 1992. Identifying potential CSCW applications by means of activity theory concepts: A case example. In *Proceedings of Computer-Supported Cooperative Work: CSCW '92*, ed. J. Turner and R. Kraut, 233–40. New York: ACM.

Kyng, M. 1995. Creating contexts for design. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 85–107. New York: John Wiley & Sons.

Lee, J. C., D. S. McCrickard, and K. T. Stevens. 2009. Examining the foundations of agile usability with extreme scenario-based design. In *Proceedings of the 2009 Agile Conference (August 24–28, 2009)*, 3–10. Washington, DC: AGILE. IEEE Computer Society. DOI= http://dx.doi.org/10.1109/AGILE.2009.30.

Lévi-Strauss, C. 1967. *Structural Anthropology*. Garden City, NY: Anchor Books.

MacLean, A., R. M. Young, and T. P. Moran. 1989. Design rationale: The argument behind the artifact. In *Proceedings of Human Factors in Computing Systems: CHI '89*, 247–52. New York: ACM.

Madsen, K. H. 1994. A guide to metaphorical design. *Commun ACM* 37(12):57–62.

Maiden, N., C. Ncube, S. Kamali, N. Seyff, and P. Grunbacher. 2007. Exploring scenario forms and ways of use to discover requirements on airports that minimize environmental impact. In *15th IEEE International Requirements Engineering Conference (RE 2007)*, 29–38. Washington, DC: IEEE Computer Society Press.

McCall, R. 2010. Critical conversations: Feedback as a stimulus to creativity in software design. *Hum Technol Interdiscip J Hum ICT Environ* 6(1):11–37.

McKerlie, D., and A. MacLean. 1994. Reasoning with design rationale: Practical experience with design space analysis. *Des Stud* 15:214–26.

Medin, D. L., and M. M. Schaffer. 1978. A context theory of classification learning. *Psychol Rev* 85:207–38.

Moran, T., and J. M. Carroll. eds. 1996. *Design Rationale: Concepts, Techniques, and Use*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Muller, M. 1991. PICTIVE—An exploration in participatory design. In *Proceedings of Human Factors in Computing Systems*: CHI '91, 225–31. New York: ACM.

Muller, M. K. 1992. Retrospective on a year of participatory design using the PICTIVE technique. In *Proceedings of Human Factors of Computing Systems: CHI '92*, ed. A. Janda, 455–62. New York: ACM.

Muller, M. J., L. G. Tudor, D. M. Wildman, E. A. White, R. A. Root, T. Dayton, R. Carr, B. Diekmann, and E. Dykstra-Erickson. 1995. Bifocal tools for scenarios and representations in participatory activities with users. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 135–63. New York: John Wiley & Sons.

Mylopoulos, J., L. Chung, and B. Nixon. 1992. Representing and using nonfunctional requirements: A process-oriented approach. *IEEE Transact Softw Eng* 18(6):483–97.

Nardi, B. A. ed. 1996. *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, MA: MIT Press.

Nielsen, J. 1995. Scenarios in discount usability engineering. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 59–83. New York: John Wiley & Sons.

Nielsen, J., and R. L. Mack. 1994. *Usability Inspection Methods*. New York: John Wiley & Sons.

Norman, D. A. 1986. Cognitive engineering. In *User Centered System Design*, ed. D. A. Norman and S. D. Draper, 31–61. Hillsdale, NJ: Lawrence Erlbaum Associates.

Norman, D. A. 1988. *The Psychology of Everyday Things*. New York: Basic Books.

Norman, D. A. 2005. Human-centered design considered harmful. *Interactions* 12(4):14–8.

Orr, J. 1986. Narratives at work: Story telling as cooperative diagnostic activity. In *Proceedings of CSCW '96*. 62–72. New York: ACM.

Payne, C., C. F. Allgood, C. M. Chewar, C. Holbrook, and D. S. McCrickard. 2003. Generalizing interface design knowledge: Lessons learned from developing a claims library. In *Proceedings of IRI '03,* 362–9. New York: IEEE.

Polson, P. G., C. Lewis, J. Rieman, and C. Wharton. 1992. Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *Int J Man Mach Stud* 36:741–73.

Pommeranz, A., W. Brinkman, P. Wiggers, J. Broekens, and C. M. Jonker. 2009. Design guidelines for negotiation support systems: An expert perspective using scenarios. In *European*

Conference on Cognitive Ergonomics: Designing Beyond the Product—Understanding Activity and User Experience in Ubiquitous Environments (Helsinki, Finland, September 30–October 02, 2009). ed. L. Norros, H. Koskinen, L. Salo, and P. Savioja, 1–8. Finland: European Conference on Cognitive Ergonomics. VTT Technical Research Centre of Finland, VTT.

Potts, C. 1995. Using schematic scenarios to understand user needs. In *Proceedings of ACM Symposium on Designing Interactive Systems: DIS'95: (Ann Arbor, Michigan)*, 247–56. New York: ACM Press.

Propp, V. 1958. *Morphology of the Folktale*. Mouton: The Hague (originally published in 1928).

Rogers, Y. 2009. The changing face of human-computer interaction in the age of ubiquitous computing. In *Proceedings of the 5th Symposium of the Workgroup Human-Computer interaction and Usability Engineering of the Austrian Computer Society on HCI and Usability For E-Inclusion (Linz, Austria, November 09–10, 2009)*, Lecture Notes in Computer Science, ed. A. Holzinger and K. Miesenberger, Vol. 5889, 1–19. Berlin, Heidelberg: Springer-Verlag. DOI= http://dx.doi.org/10.1007/978-3-642-10308-7_1.

Rosch, E., C. B. Mervis, W. Gray, D. Johnson, and P. Boyes-Braem. 1976. Basic objects in natural categories. *Cogn Psychol* 7:573–605.

Rosson, M. B. 1999. Integrating development of task and object models. *Commun ACM* 42(1):49–56.

Rosson, M. B., and J. M. Carroll. 1993. Extending the task-artifact framework. In *Advances in Human-Computer Interaction*, ed. R. Hartson and D. Hix, Vol. 4, 31–57. New York: Ablex.

Rosson, M. B., and J. M. Carroll. 1995. Narrowing the gap between specification and implementation in object-oriented development. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 247–78. New York: John Wiley & Sons.

Rosson, M. B., and J. M. Carroll. 1996. The reuse of uses in Smalltalk programming. *ACM Trans Comput Hum Interact* 3(3):219–53.

Rosson, M. B., and J. M. Carroll. 2000a. Nonfunctional requirements in scenario-based development. In *Proceedings of OZCHI 2000*, 232–9. North Ryde, NSW, Australia: CSIRO Mathematical and Information Sciences.

Rosson, M. B., and J. M. Carroll. 2000b. Scenarios, objects, and points-of-view in user interface design. In *Object Modeling and User Interface Design*, ed. M. van Harmelen. London: Addison-Wesley Longman.

Rosson, M. B., and J. M. Carroll. 2002. *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. San Francisco, CA: Morgan Kaufmann.

Rosson, M. B., and E. Gold. 1989. Problem-solution mapping in object-oriented design. In *Proceedings of OOPSLA'89: Conference on Object-Oriented Programming Systems, Languages, and Applications*, ed. N. Meyrowitz, 7–10. New York: ACM.

Rosson, M. B., S. Maass, and W. A. Kellogg. 1988. The designer as user: Building requirements for design tools from design practice. *Commune ACM* 31:1288–98.

Rubin, K., and A. Goldberg. 1992. Object behavior analysis. *Commun ACM* 35(9):48–62.

Schön, D. A. 1967. *Technology and Change: The New Heraclitus*. New York: Pergamon Press.

Schön, D. A. 1983. *The Reflective Practitioner: How Professionals Think in Action*. New York: Basic Books.

Schön, D. A. 1987. *Educating the Reflective Practitioner*. San Francisco, CA: Jossey-Bass.

Scriven, M. 1967. The methodology of evaluation. In *Perspectives of Curriculum Evaluation*, ed. R. Tyler, R. Gagne, and M. Scriven, 39–83. Chicago: Rand McNally.

Sommerville, I. 1992. *Software Engineering*. Fourth ed. Reading, MA: Addison-Wesley.

Sutcliffe, A. 2002. *The Domain Theory: Patterns for Knowledge and Software Reuse*. Hillsdale, NJ: Erlbaum Associates.

Sutcliffe, A. 2010. Juxtaposing design representations for creativity. *Hum Technol Interdiscip J Hum ICT Environ* 6(1):38–54.

Sutcliffe, A. G., and S. Minocha. 1998. Scenario-based analysis of non-functional requirements. In *Proceedings of RESFQ 2000*. Namur, Belgium: Namur University Press.

Swanson, E., K. Sato, and J. Gregory. 2009. Exploring cultural context using the contextual scenario framework. In *Proceedings of the 3rd International Conference on Internationalization, Design and Global Development: Held As Part of HCI International 2009 (San Diego, CA, July 19–24, 2009)*, Lecture Notes In Computer Science, ed. N. Aykin, Vol. 5623, 117–26. Springer-Verlag, Berlin: Heidelberg. DOI= http://dx.doi.org/10.1007/978-3-642-02767-3_13.

Theng, Y. L., D. H. Goh, E. Lim, Z. Liu, N. L. Pang, P. B. Wong, and L. Chua. 2002. Intergenerational Partnerships in the Design of a Digital Library of Geography Examination Resources. In *Proceedings of the 5th International Conference on Asian Digital Libraries: Digital Libraries: People, Knowledge, and Technology (December 11–14, 2002)*, Lecture Notes In Computer Science, ed. E. Lim, S. Foo, C. S. Khoo, H. Chen, E. A. Fox, S. R. Urs, and C. Thanos, Vol. 2555, 427–39. London: Springer-Verlag.

Tideman, M., M. C. van der Voor, and B. van Arem. 2010. A new scenario based approach for designing driver support systems applied to the design of a lane change support system. *Transp Res Part C Emerg Technol* 18(2):247–58.

Turner, P., and S. Turner. 2011. Is stereotyping inevitable when designing with personas? Design Studies 11(3):30–44.

Tversky, A, and D. Kahneman. 1974. Judgements under uncertainty: Heuristics and biases. *Science* 185:1124–31.

Vasara, K. 2003. *Introducing Personas in a Software Development Project*. Masters thesis submitted to Helsinki University of Technology. Department of Computer Science & Engineering, Helsinki, Finland.

Verplank, W. L. 1988. Graphic challenges in designing object-oriented user interfaces. In *Handbook of Human-Computer Interaction*, ed. M. Helander, 365–76. Amsterdam: North-Holland.

Virzi, R. A., J. L. Sokolov, and D. Karis. 1996. Usability problem identification using both low- and high-fidelity prototypes. In *Proceedings of Human Factors in Computing Systems: CHI '96*, 236–43. New York: ACM.

Wasserman, A. I., and D. T. Shewmake. 1982. *Rap Prototyping Interact Inf Syst ACM Softw Eng Notes* 7(5):171–80.

Weidenhaupt, K., K. Pohl, M. Jarke, and P. Haumer. 1998. Scenarios in system development: Current practice. *IEEE Softw* 15/2:34–45.

Wertheimer, M. 1938. Laws of organization in perceptual forms. *A Sourcebook of Gestalt Psychology*, ed. W. D. Ellis. London: Paul, Trench, Trubner.

Wirfs-Brock, R. 1995. Designing objects and their interactions: A brief look at responsibility-driven design. In *Scenario-Based Design: Envisioning Work and Technology in System Development*, ed. J. M. Carroll, 337–60. New York: John Wiley & Sons.

Wirfs-Brock, R., and Wilkerson, B. 1989. Object-oriented design: A responsibility-driven approach. In *Object-Oriented*

Programming: Systems, Languages and Applications, Proceedings of OOPSLA'89, 71–6. New York: ACM.

Wirfs-Brock, R., B. Wilkerson, and L. Wiener. 1990. *Designing Object-Oriented Software*. Englewood Cliffs, NJ: Prentice Hall.

Young, R. M., and P. B. Barnard. 1987. The use of scenarios in human-computer interaction research: Turbocharging the tortoise of cumulative science. In *Proceedings of CHI+GI'87: Conference on Human Factors in Computing Systems and Graphics Interface (Toronto, Canada, April 5–9)*, 291–6. NewYork: ACM.

Zhao, D., and M. B. Rosson. 2009. How and why people twitter: The role that micro-blogging plays in informal communication at work. In *Proceedings of GROUP 2009*, 243–52. New York: ACM.

49 Participatory Design

The Third Space in Human–Computer Interaction

Michael J. Muller and Allison Druin

CONTENTS

49.1 INTRODUCTION: JUST ADD USERS AND STIR?

In a discussion of integrating women's perspectives into a male-dominated curriculum, Bunch (1987) noted that "you can't just added women and stir" (p. 140). It takes work, and new ways of thinking, and new kinds and methods of openness to bring substantively new voices into a conversation. Similarly, to bring users' knowledges and perspectives directly into computer specification and design, it is necessary to do more than "just add users and stir." This chapter surveys methods that go beyond merely adding users—methods to create new settings and experiences that can assist computer professionals to work in partnership with diverse users in improving both computer technology and the understandings that make computer technologies successful in real use.

Participatory design (PD) is a set of theories, practices, and studies related to end users as full participants in activities leading to software and hardware computer products and computer-based activities (Greenbaum and Kyng 1991; Muller and Kuhn 1993; Schuler and Namioka 1993). The field is extraordinarily diverse, drawing on fields such as user-centered design, graphic design, software engineering, architecture, public policy, psychology, anthropology, sociology, labor studies, communication studies, and political science and from localized experiences in diverse national and cultural contexts (Gregory 2003). This diversity has not lent itself to a single theory or paradigm of study or approach to practice (Beck 1996; Bjerknes and Bratteteig 1995; Clement and Van den Besselaar 1993; Kensing and Blomberg 1998a; Slater 1998; Suchman 2002). Researchers and practitioners are brought together—but are not necessarily brought into unity—by a pervasive concern for the knowledges, voices, and/or rights of end users, often within the context of technology design and development, or of other institutional settings (e.g., workers in companies, corporations, universities, hospitals, and governments) (Bødker 1990; Bødker et al. 1988; Gregory 2003) or of other experiences in life (e.g., children, older adults, and people with disabilities) (Druin 2002; Guha and Druin 2008; Hornof 2008; Xie et al. in press; see also Chapter 40). Many researchers and practitioners in PD (but not all) are motivated in part by a belief in the value of democracy to civic, educational, and commercial settings—a value that can be seen in the strengthening of disempowered groups including workers, children, and older adults, in the improvement of internal processes, and in the combination of diverse knowledge to make better services and products (Beyer and Holtzblatt 1998; Béguin 2003; Bjerknes, Ehn, and King 1987; Braa 1996; Briefs, Ciborra, and Schneider 1983; Bødker, Kensing, and Simonsen 2004; Carroll 1995, 2000; Checkland 1981; Clement, Kolm, and Wagner 1994; Docherty et al. 1987; Druin 2002; Ehn 1988, 1993; Floyd 1993; Floyd et al. 1989; Gasson 1995; Gregory 2003; Greenbaum and Kyng 1991; Kensing and Blomberg 1998b; Klær and Madsen 1995; Kyng and Matthiessen 1997; Madsen 1999; McLagan and Nel 1995; Muller and Kuhn 1993; Mumford 1983; Mumford and Henshall 1979/1983; Noro and Imada 1991; Nygaard 1975; Scrivener, Ball, and Woodcock 2000; Schuler and Namioka 1993; Spencer 1989; Suchman 1995, 2002; Van den Besselaar, Clement, and Jaervinen 1991; Xie et al. in press; Wixon and Ramey 1996).

PD began in an explicitly political context, as part of the Scandinavian workplace democracy movement (e.g., Nygaard 1975; Bjerknes, Ehn, and Bratteteig 1987; Ehn and Kyng 1987; Floyd et al. 1989; more recently, see Bjerknes and Bratteteig [1995]; Beck [1996, 2001]; Gregory [2003]; Kyng and Matthiessen [1997]; Winner [1994]). Early work took the form of experiments conducted by university researchers in alliances with organized labor (for historical overviews, see Ehn [1993]; Gregory [2003]; Levinger [1998]). More recent work has more explicitly considered additional social justice issues such as inclusive design (Light and Luckin 2008), women's needs (Balka 1995; Greenbaum 1991; Nisonen 1994), cultural sensitivity (Druin et al. 2009; Kam et al. 2006), disabilities challenges (Hornof 2008), and more general issues of exclusion related to race, age, gender, and/or class (DiSalvo et al. 2010; Druin 2002).

Subsequent work focused on combining complex and distinct knowledges for realistic design problems. Segalowitz and Brereton (2009) described three attributes of new knowledge that could lead to difficulties in participation: novelty, difference, and dependence. Winters and Mor (2008) discussed the need for a methodology of interdisciplinary knowledge exchanges to support participation design. Fowles (2000) wrote of transforming the "symmetry of ignorance" (mutual incomprehension between designers and users) into a complementary "symmetry of knowledge" through symmetries of participation and symmetries of learning. Nielsen and Bødker (2009) recently updated this analysis for the current context of virtual collaborations with users. Similarly, Holmström (1995) analyzed a "gap in rationalities" among developers and users, and Béguin (2003) argued for the need to close this gap through mutual learning among designers and end users. Reymen, Whyte, and Dorst (2005) considered the diverse knowledges that are needed in design (see also Badke-Schaub [2004]), and Louridas (1999) provided an influential analysis of the similar thought patterns that are used with different conceptual vocabularies by professional versus nonprofessional designers. In view of these different conceptual vocabularies, one of us wrote about the need for translations among the coequal worlds of users and software professionals and the need to foster a polyvocal polity in which these various interested parties could coconstruct new concepts, meanings, and alliances (Muller 1997a, 1997b). Suchman (2002) described her historical practice of PD as "working for the presence of multiple voices not only in knowledge production, but in the production of technologies as knowledges objectified in a particular way." Bødker and Buur (2002) noted the need to support the "many-voiced nature of design." These acknowledgments of the integrity and rationality of multiple voices and multiple knowledges (e.g., users and software professionals) are a crucial aspect of the argument of this chapter, concerning the creation of hybrid spaces between and among those diverse perspectives.

However, the integrity of including multiple voices in design has been questioned. Reyman et al. (2005) summarized the problem from the perspective of professional designers, whose newly won strength in systems' design is challenged by the claims of users' knowledge as a crucial component of design. They noted that "designers have their own expertise," and "it is not yet clear which kind of user involvement is most appropriate." Luck (2003) explored issues of disagreement, even among the users. Druin, this chapter's second author, suggested that there are four roles children can play in the design process: user, tester, informant, and design partner (Druin 2002). With each role there is a spectrum of user involvement, at differing points in the design of new technology. Jönsson and colleagues (n.d.) listed a series of design constraints for working with seniors (see also Demirbilek and Demirkan [2004]). Yamauchi (2009) suggested that best role for users was as "peripheral designers," working with assigned detailed problems rather than whole-system design. Light and Luckin called into question a simplified view of involving everyone in design projects without methods and techniques to enfranchise diverse participants:

> "Believing in the potential of everyone to design is more egalitarian than believing in exclusive talents and specialized roles. However, this is not the same as involving every potential user in every design project, or at all stages, or in the same way as the next person." (Light and Luckin 2008, p. 16)

In effect, the observation by Light and Luckin returns us to our opening theme, "you can't just 'add users and stir.'" People's needs differ by work roles and their relationship to the design task, by life stage, by physical or cognitive condition, and by other attributes and dimensions as well. People need different design affordances and degrees of safety, depending on their circumstances, their identities, and their relationship to the design task and its social or organizational setting. These issues help to motivate this chapter's survey of participatory methods, and particularly our focus on new "hybrid" spaces for mutual learning and reciprocal validation of diverse perspectives.

Recently, PD has achieved a status as a useful commercial tool in some settings (e.g., McLagan and Nel 1995), with several major and influential consultancies forming their business identities around participatory methods,* and an increasing number of textbooks for design or IT governance based on participatory principles (Beyer and Holtzblatt 1998; Bødker, Kensing, and Simonsen 2004). This overall corporate and managerial "mainstreaming" of PD has been greeted by some with enthusiasm and by others with dismay. Participatory work in the United States has sometimes been criticized as too friendly to management or too limited by the users' experience. Participatory work on the Pacific Rim (e.g., Noro and Imada 1991) appears to have grown out of the quality movement and focuses much more on solving problems and much less on changing workplace power relations. On the other hand, PD has gained growing acceptance in the world of research, particularly from academic professionals in Europe and North America focused on developing new technologies for children (e.g., Druin 1999/2002; Garzotto 2008; Hornof 2008; Jones et al. 2003; Kam et al. 2006; Large et al. 2007; Mazzone, Read, and Beale 2008; Robertson 2002; Taxén 2004). Adapting the notions of changing the "power structures," researchers have sought to give children a voice in the design of new technologies with the belief that more appropriate solutions can be found.

Historically, as summarized by Gregory (2003; see also Kensing and Blomberg [1998a]), PD has included both a "conflict-perspective," such as the collective resource tradition (Ehn and Kyng 1987), as well as approaches that are more integrated into conventional work processes (e.g., Beyer and Holtzblatt 1998; Noro and Imada 1991; Sanders 2000; and perhaps Bødker, Kensing, and Simonsen 2004). The integrationist approaches (including those practiced by one of us) have been critiqued as an insufficient "harmony perspective" by, for example, Ehn (1993) and Kyng (1998). Indeed, several definitions of conflict have been fruitful for PD. The concept of breakdowns in anticipated working practices was explored in an influential treatment by Bødker (1990) within the theoretical frame of activity theory. In this approach, the conflict is between expectation and initial outcomes, giving motivation and direction to a need for changes. The concept of class conflict has also been useful, especially in the Scandinavian context (Beck 1996, 2001; Bjerknes and Bratteteig 1995; Bjerknes, Ehn, and Bratteteig 1987; Gregory 2003), where it has served as the organizing principle for work with trade unions as powerful stakeholders and allies in those countries. Elsewhere, a more muted approach of identifying problems and gaps between the present and the future has informed participatory work where the labor movement is weaker (Beyer and Holtzblatt 1998; Carroll 1995, 2000; Checkland 1981; Lafreniére 1996; Muller et al. 1995b; Mumford 1983; Mumford and Henshall 1979/1983; Noro and Imada 1991), where the conflict is between history and current needs (Best et al. 2009; Cameron 1998; Carmien et al. 2003; Davies et al. 2004; Enquist and Tollmar 2008; Fowles 2000; Hirsch 2009; Moffatt et al. 2004; Nisonen 1994; Pecknold 2009; Salvador and Howells 1998; Salvador and Sato 1998; Taylor and Cheyerst 2009; Wu, Richard, and Baecker 2004; Wu, Baecker, and Richards 2005), or in projects in which the focus is on design rather than on workplace (e.g., Béguin 2003; Binder 1999; Brandt and Messter 2004; Buur, Binder, and Brandt 2000; Carter and Mankoff 2005; Dandavate, Steiner, and William 2000; Druin 1999; Druin et al. 2000/2009; Hornecker 2010; Howard et al. 2002; Iacucci, Iacucci, and Kuutti 2002; Iacucci and Kuutti 2002; Kankainen et al. 2005; Kantola et al. 2007; Kuutti, Iacucci, and Iacucci 2002; Merkel et al. 2004; Nielsen and Bødker 2009; Pedersen and Buur 2000; Sanders 2000, 2006; Sanders and Branaghan 1998; Sanders and Nutter 1994; Tschudy, Dykstra-Erickson, and Holloway 1996; Vaajakallio and Mattelmäki 2007; Wakkary and Tanenbaum 2009).

A more recent trend has been the maturing of lifecycle approaches to participatory work. Early and somewhat experimental lifecycle models were offered by Mumford (1983) and Floyd (1993), anticipated in some ways by Checkland (1981). Two more mature approaches have been offered by Beyer and Holtzblatt (1998) and Bødker, Kensing, and Simonsen (2004). A further incorporation of participatory methods into large-scale conventional processes was explored in Pew and Mavor (2007). Finally, we note that, according to some researchers in the field of end user innovation and user appropriation studies, new technologies have become so complex that ordinary users will have to modify those technologies in order to "domesticate them" and make them fit for use (e.g., Aune 1996; Cook and Light 2006; von Hippel 2002; Light and Luckin 2008; Silverstone and Haddon 1998; Wakkary and Tanenbaum 2009). This rich area of research and practice is regrettably beyond the scope of this chapter.

This chapter primarily addresses methods, techniques, and practices in PD, with modest anchoring of those practices in theory. We will not repeat our earlier encyclopedic survey of participatory practices (Muller, Haslwanter, and Dayton 1997). Rather, we will pursue a trend within those practices that has shown the most growth during the past years, and we will motivate our interest in that trend through recent advances in the domain of cultural studies. We will focus on participatory practices that fall in the hybrid realm between the two distinct work domains of (1) technology developers/researchers and (2) end users.

We should also say that our concern is for methods that have been *shown to work in real situations*—that is, that address real

* In the interest of fairness to other consultancies, we will not provide the names of commercial ventures.

problems in work life, education, home life, leisure, and so on—and in which the outcomes were of consequence, and in which the participants could freely choose whether to be involved in the work. We have therefore omitted many promising methods that have so far been explored only as in-laboratory university exercises, apparently as part of assigned coursework. Instead we look toward more realistic explorations of these new methods.

In this third edition of the *Handbook*, we have also expanded the domains in which we report participatory methods and techniques. Previous editions have focused on work and workers, usually in face-to-face settings. In our new version of this chapter, we also include participatory work with children and with people with disabilities, and we bring in methods from the emerging subfield of distributed PD (DPD) (as practiced among *non*colocated collaborators) where appropriate, and participatory methods as used in the special circumstances of the developing world. Our expanded scope may be seen as a further dilution of the labor orientation to PD. In response, we hope that this broadened sense of *who matters in design* will ultimately lead to greater enfranchisement and new alliances for change.

49.1.1 Major Bibliographic Sources for Participatory Design

Theory, practice, and experience in PD have been published in a series of conference proceedings and several major books.

49.1.1.1 Conference Series

Seven important conference series have made major contributions to PD:

- *Critical Computing.* Four conferences have been held, at 10-year intervals, in the Critical Computing series, most recently in 2005. Major papers from the conferences have appeared as two influential books (Bjerknes, Ehn, and Bratteteig 1987; Kyng and Matthiessen 1997).
- *Information Systems Research in Scandinavia (IRIS) Conference.* The annual IRIS conference series often include sessions and individual contributions on participatory topics. *Proceedings* may be available through the IRIS Association or online.*
- *PD Conference.* The PD Conference has met on even-numbered years since 1990. Earlier *Proceedings* were published by computer professionals for social responsibility (CPSR);† more recent *Proceedings* were published by the Association for Computing Machinery.‡ Selected papers from several conferences have appeared in edited volumes or special journal issues (e.g., Kensing and Blomberg 1998a; Muller and Kuhn 1993; Schuler and Namioka 1993). Papers from recent conference years are available through the ACM Digital Library.§
- *Include Conferences.* The Helen Hamlyn Center¶ has sponsored a series of conferences on inclusive design since 2003, and provides additional materials in this area. The concept of inclusive design emphasizes enfranchising as broad a range of people as possible, usually with a focus on removing barriers related to physical, cognitive, and emotional disabilities. The Include Conferences have explicitly included emphases on home, civic life, and *workplace* within this broader agenda, and with themes of participatory work with people of diverse backgrounds and abilities.
- *IFIP Conferences.* A number of conferences and workshops (sponsored by IFIP Technical Committee (TC) 9) have focused on selected topics within PD—for example, Briefs, Ciborra, and Schneider (1983); Clement, Kolm, and Wagner (1994); Docherty et al. (1987); Gärtner and Wagner (1995); and van den Besselaar et al. (1991).**
- *Nordic Conferences on Human–Computer Interaction.* The Nordic Conferences on Human–Computer Interaction (NordiCHI) conference series meets on even-numbered years, with a strong emphasis on participatory work within a broader Scandinavian context (Nordichi 2006). Papers from 2002 and 2004 are available through the ACM Digital Library.
- Major papers, panels, and tutorials on PD have also appeared in the CHI, CSCW, ECSCW, and DIS conference series, beginning as early as 1988 (*Proceedings* available through the Association for Computing Machinery or through Springer for the ECSCW conference series), and in *Proceedings* of the Usability Professionals' Association†† conference series, of the INTERACT conference series, and of the Human Factors and Ergonomics Society conference series. Several papers at the Co-Designing 2000 Conference‡‡ addressed participatory themes (Scrivener, Ball, and Woodcock 2000).
- *Interaction Design and Children (IDC).* From this yearly conference's inception in 2002, researchers worldwide have published and presented papers where a surprising number discuss design methods that are inclusive of children in the development/research process (*Proceedings* are also available through the Association for Computing Machinery§§). With conference venues routinely in both Europe and the United States, a strong Scandinavian influence has been seen with the embracing of PD methods in this research area concerning children.

* http://iris.informatik.gu.se/.

† www.cpsr.org.

‡ www.acm.org.

§ http://portal.acm.org/dl.cfm.

¶ http://www.hhc.rca.ac.uk/.

** http://www.ifip.or.at/. For TC 9, see http://www.ifip.or.at/bulletin/bulltcs/memtc09.htm.

†† www.upassoc.org.

‡‡ http://vide.coventry.ac.uk/codesigning/.

§§ www.acm.org.

49.1.1.2 Books

In addition to the books cited above, major collections of papers and/or chapters related to PD appeared in Carroll's volume on scenarios in user interaction (1995; see also Carroll [2000]), Greenbaum's and Kyng's *Design at Work* (1991), and Wixon's and Ramey's collection of papers on field-oriented methods (1996). Individual books that have been influential in the field include Bødker's application of activity theory to issues of participation (1990), Ehn's account of work-oriented design (1988), Suchman's discussion of situated action (1987), and Beyer's and Holtzblatt's presentation of contextual inquiry and contextual design (1998; see also Chapter 43 in this book). A recent volume by Bødker, Kensing, and Simonsen (2004) may broaden the impact of PD among information technology departments.* Earlier influential works include a series of books on socio-technical theory and practice by Mumford (e.g., 1983; Mumford and Henshall 1979/1983), as well as Checkland's (1981) soft systems methodology. Noro and Imada (1991) developed a hybrid ergonomic approach, involving participation and quality programs, which has been influential around the Pacific Rim. For a historical PD bibliography, see the CPSR website.

49.1.1.3 Journals

Three journals have carried the greatest number of PD papers: Scandinavian Journal of Information Systems,† Computer Supported Cooperative Work: The Journal of Collaborative Computing,‡ and Human–Computer Interaction.§

49.1.1.4 Websites

CPSR maintains a set of PD resources at http://www.cpsr.org/issues/pd/.

The group of researchers working on DPD, or participation at a distance, has created a website that includes *Proceedings* from their 2006 and 2008 conference workshops, at http://extra.shu.ac.uk/paperchaste/dpd/index.html.

49.2 HYBRIDITY AND THIRD SPACE

This chapter is concerned with participatory methods that occur in the hybrid space between technology developers/researchers and end users. Why is this hybrid space important?

Bhabha (1994) made an influential argument that the border or boundary region between two domains—two spaces—is often a region of overlap or hybridity—that is, a "third space" that contains an unpredictable and changing combination of attributes of each of the two bordering spaces. His area of concern was colonization, in which some native people find themselves caught in between their own traditional culture and the newly imposed culture of the colonizers (see also Dingawaney and Maier [1994]; Karttunen [1994]). Their continual negotiation and creation of their identities, as efforts of survival, creates a new hybrid or third culture (Bhabha 1994; see also Lyotard [1984]) and even a third language (Anzaldúa 1999; Bachmann-Medick 1996). In such a hybrid space, enhanced knowledge exchange is possible, precisely because of those questions, challenges, reinterpretations, and renegotiations (Bachmann-Medick 1996). These dialogues across differences and—more importantly—*within differences* are stronger when engaged in by groups, emphasizing not only a shift from assumptions to reflections, but also from individuals to collectives (Carrillo 2000). Bhabha's conception has become highly influential. Bachmann-Medick (1996) applied the concepts to translation theory. Grenfell (1998) interpreted concepts of hybridity in a study of living-at-the-border in multicultural education settings. Evanoff (2000) surveyed a number of theoretical applications of hybridity, from evolutionary biology to constructivist perspectives in sociology to democratic responses to intercultural ethical disagreements. He explored formulations from multiple disciplines, involving "third culture" in intercultural ethics, "third perspective" involving "dynamic inbetweenness" in Asian-Western exchanges, and a psychological "third area" in the development of a "multicultural personality." A summary of the claims relating to third spaces (or hybridity) appears in Table 49.1.

TABLE 49.1
Summary of Claims Relating to Third Spaces

Overlap between two (or more) different regions or fields (inbetweenness)
Marginal to reference fields
Novel to reference fields
Not "owned" by any reference field
Partaking of selected attributes of reference fields
Potential site of conflicts between/among reference fields
Questioning and challenging of assumptions
Mutual learning
Synthesis of new ideas
Negotiation and (co)creation of…
Identities
Working language
Working assumptions and dynamics
Understandings
Relationships
Collective actions
Dialogues across and within differences (disciplines)
Polyvocality
What is considered to be data?
What are the rules of evidence?
How are conclusions drawn?
Reduced emphasis on authority – increased emphasis on interpretation
Reduced emphasis on individualism – increased emphasis on collectivism
Heterogeneity as the norm

* In addition, Pew and Mavor (2007) included participatory design among their proposed "new look" at large-systems development. However, the influence of this work has not yet been determined.

† http://www.cs.auc.dk/~sjis/

‡ http://www.wkap.nl/journalhome.htm/

§ http://hci-journal.com/

49.2.1 Hybridity and HCI

Within HCI, Suchman recently renewed her call for dialogue across boundaries between the partial perspectives of end users and developers (Suchman 2002; see also Badke-Schaub [2004]; Bødker and Buur [2002]; Fowles [2000]; Holmström [1995]; Kyng [1998]; Light and Luckin [2008]; Nygaard and Sørgaard [1987]). Suchman argued for boundary crossing and mutual learning between these different standpoints and appealed in part to recent developments in feminist epistemologies which argue that objectivity is the constructive outcome of an ongoing dialogue among multiple perspectives (e.g., Haraway 1991; Harding 1991; Hartsock 1983; see also Brereton [2009]). These concerns become more pressing when we consider the new obstacles encountered in DPD (Naghsh et al. 2008), especially when design work also spans the boundaries between the developed world and the developing world (e.g., Best et al. 2009; Bidwell and Hardy 2009; Bidwell et al. 2010). Titlestad, Staring, and Braa (2009) explained:

> “A key PD principle is to bridge and blur the user-designer distinction from both directions, through mutual learning processes .… Effective methods to achieve this usually rely on prototyping and intensive face-to-face iteration.… In the Global South, computerized information systems are still few and far between…a significant threshold hindering participation.…”

In partial agreement with Suchman, Warr (2006) argues that the solution is not to remove distance entirely, but rather to *preserve the situated nature of each participant’s own world* while creating a common space for mutual learning, creation, and problem solving.

The approach in this chapter begins with a similar recognition of diverse perspectives. However, unlike Suchman’s and Titlestad et al.’s emphasis on the boundary *between* these perspectives, this chapter is concerned with creating regions of overlap where the perspectives can come into mutual knowledge and, potentially, alliance—with the creation of the hybrid spaces in which objectivity can emerge through constructive discussion, dialogue, negotiation, and mutual learning. Similarly, this chapter pursues a different solution from the located accountability recommended by Suchman. Suchman sees each participant as located *within* a particular perspective and interest—for example, “Organizations comprise multiple constituencies each with their own professional identities and views of others” (see also the geographic limits discussed by Titlestad, Staring, and Braa [2009]). By contrast, the methods in this chapter enable the creation of new perspectives and new locations and acknowledge the possibility that each participant can make different choices at different moments about where to locate her or his perspective, standpoint, and thus accountability. In keeping with the origins of PD in class struggle (e.g., Ehn and Kyng 1987; see also Gregory [2003], for a review of “conflict-perspective” approaches), Suchman focuses on opposing interests that meet across a designated boundary. This chapter proposes to reach toward the next step—i.e., to pursue the polyvocal polity that one of us proposed (Muller 1997a) and the need identified by Bødker and Buur (2002; see also Buur and Bødker [2000]) to create a “meeting ground” for a “widen[ed]… circle of participants” that can “support the many voices being brought forth in order to create the new, and to find ways of supporting this multivoicedness.”

There have been many calls within HCI for mutual or reciprocal learning in hybrid spaces (e.g., Bødker et al. 1987, 1988; Druin 1999/2002; Druin et al. 2000; Ehn and Sjögren 1991; Floyd 1987; Kensing and Madsen 1991; Lanzara 1983; Mogensen and Trigg 1992; Muller 1997a; Muller, Wildman, and White 1994; Mumford 1983; Törpel and Poschen 2002; Tscheligi et al. 1995). Beeson and Miskelly (2000) appealed to the notion of hybridity (“heterotopia”) in describing workers who, like colonized peoples, deal “in a space which is not their own,” (p. 2) taking limited and opportunistic actions to preserve “plurality, dissent, and moral space” (p. 1). Maher, Simoff, and Gabriel (2000) described the creation of virtual design spaces for sharing diverse perspectives. Merkel et al. (2004) described a need for “a new set of skills and competencies that go beyond technical design skills … to create conditions that encourage a collaborative design process and active reflection … for working with groups … that push on the traditional boundaries between users and designers” (p. 7–8). Light and Luckin (2008) discussed hybrid methods of enfranchisement for people with diverse backgrounds. In an early formulation, Lanzara (1983) suggested that

> [A] large part of the design process, especially in large-scale projects and organizations involving several actors, is not dedicated to analytical work to achieve a solution but mostly to efforts at reconciling conflicting [conceptual] frames or at translating one frame into another. Much work of the designer is … concerned with … defining collectively what is the relevant problem, how to see it.

Tscheligi et al. (1995), in a panel on prototyping, considered that the “products” of prototyping include not only artifacts, but also understandings, communications, and relationships—a theme that was echoed in a more recent panel on modeling (Kaindl et al. 2001). Fanderclai (1995, 1996) captured a strong sense of possible new dynamics and new learnings in a hybrid online space. Finally, Thackara (2000) based part of his plenary address at CHI 2000 on the concept of the third space, providing a needed hybridity to HCI studies.

49.2.2 Participatory Design as the Third Space in HCI

In this chapter, we extend the HCI analyses surveyed in the preceding paragraphs and apply Bhabha’s perspective to the HCI problem of methods to bridge between two spaces—the world of technology developers/researchers and the world of the end users (see also Muller [1997a, b]). As noted by Suchman (2002), each world has its own knowledges and practices; each world has well-defined boundaries. Movement from one world to the other is known to be difficult (Dewulf and Van Meel 2003; Kensing and Blomberg 1998a; Kujala 2003; Luck 2003; Olsson

2004; Reymen, Whyte, and Dorst 2005; Yamauchi 2009). We can see this difficulty manifested in our elaborate methods for requirements analysis, design, and evaluation—and in the frequent failures to achieve products and services that meet users' needs and/or are successful in the marketplace.

Much of the traditional scientific practice in HCI has focused on instruments and interventions that can aid in transferring information between the users' world and the software world. Most of the traditional methods are relatively one directional—for example, we analyze the requirements *from* the users; we deliver a system *to* the users; we collect usability data *from* the users. Although there are many specific practices for performing these operations, relatively few of them involve two-way discussions, and fewer still afford opportunities for the software professionals to be surprised—that is, *to learn something that we didn't know we needed to know.*

The PD tradition has, from the outset, emphasized mutuality and reciprocity—often in a hybrid space that enabled new relationships and understandings. Bødker et al. (1988) made specific references to "the mutual validation of diverse perspectives" (see also Badke-Schaub [2004]; Béguin [2003]; Bødker and Buur [2002]; Fowles [2000]; Holmström [1995]; Kyng [1998]; Light and Luckin [2008]; Louridas [1999]; Reymen, Whyte, and Dorst [2005]; Suchman [2002]). Floyd (1987) analyzed software practices into two paradigms, which she termed product oriented (focused on the computer artifact as an end in itself) and process oriented (focused on the human work process, with the computer artifact as means to a human goal). In her advocacy of balancing these two paradigms, Floyd noted that the process-oriented paradigm required mutual learning among users and developers (see also Segall and Snelling [1996]). Most of the PD theories and practices require the combination of multiple perspectives—in part, because complex human problems require multiple disciplines (e.g., software expertise and work-domain expertise) for good solutions (e.g., Pew and Mavor 2007, 2000; Holmström 1995) and in part because the workplace democracy tradition reminds us that all of the interested parties (in the United States, we would say "stakeholders") should have a voice in constructing solutions (e.g., Ehn and Kyng 1987; Kyng 1998). In a related development, there are increasing calls for critical reflection in design, based on combining perspectives across disciplines, including the recent Aarhus Conference on Critical Computing.

Finally, the hybridity theme of novelty and creativity is echoed in participatory goals and practices. PD has often emphasized change—change in technology, change in working practices, and change in working relationships (Gregory 2003; Kensing and Blomberg 1998a). The earliest projects such as DEMOS, DUE, FLORENCE, and UTOPIA were concerned with anticipating and co-determining change that was mandated for various workplaces (Ehn and Sanberg 1979; Kyng and Mattiassen; Bjerknes and Bratteteig 1987; Bødker et al. 1987). Some of this early work took a critical stance with regard to managerial agendas; other projects specifically explored alternative designs (Bjerknes and Bratteteig 1987; Bødker et al. 1987; Ehn 1993), and more recent work (detailed below) is more directly concerned with *creating* new alternatives. This is very much the case in the extensive "codesign" work of researchers and children (Druin et al. 2009). Many of the participatory projects—and even the names of the methods—reflect an orientation toward the future—for example, future workshops (Jungk and Mullert 1987) "evoking the future" (Brandt and Grunnet 2000), "anticipating future behavior of office workers" (de Jong, Kouprie and De Bruyne 2009), "hands-on the future" (Ehn and Kyng 1991), "envisioning future practices" (Vaajakallio and Mattelmäki 2007), and "evaluation of future concepts" (Hultcrantz and Ibrahim 2002). Through careful control of design attributes such as clarity and ambiguity, formality and informality, and the judicious use of different disciplinary languages, PD practitioners create new hybrid spaces to encourage innovation and to support creativity.

49.2.3 Participatory Design Contains Its Own Third Space

The preceding argument—that PD serves as a kind of third space to HCI—might be interesting, but is hardly worth a chapter in a handbook. We now turn to the question of hybridity in methods within the field of PD itself. In their "tools for the toolbox" approach, Kensing and Munk-Madsen (1993) developed a taxonomy to analyze 30 participatory methods (see also Kensing, Simonsen, and Bødker [1996]; and, for independent convergences on the same attribute, see Gjersvik and Hepsø [1998]; Luck [2000]; Reid and Reed [2000]). The first dimension of their taxonomy contrasted *abstract* methods (suitable for a software professional's organization) with *concrete* methods (suitable for work with end users).* Muller, White, and Wildman (1993) and Muller, Hallewell Haslwanter, and Dayton (1997) elaborated on this taxonomic dimension by asking *whose work domain served as the basis for the method* (in the United States, we would call this a matter of "turf," as in "on whose turf did the work take place?"). At the *abstract* end of the continuum, the users have to enter the world of the technology developers/researchers in order to participate—for example, rapid prototyping (Grønbæk 1989) and quality improvement (Braa 1996). At the *concrete* end of the continuum, the technology developers/researchers have to enter the world of the users in order to participate—for example, ethnography (Blomberg et al. 1993; Crabtree 1998; Orr and Crowfoot 1992; Suchman and Trigg 1991; see also Chapter 45 in this book), ongoing tailoring during usage (Henderson and Kyng 1991; MacLean et al. 1990), and end-user "design" by purchasing software for small companies (Krabbel and Wetzel 1998; Robertson 1996, 1998). For the purposes of this chapter, we can now ask: What about the practices that did not occur at the *abstract* or *concrete* end points of the continuum? *What about the practices in between?* These practices turn out to occur in an uncertain, ambiguous, overlapping disciplinary domain that does not "belong" to either the technology

* Their second dimension was of less interest for the purposes of this chapter.

developer/researcher or the end users (i.e., these practices occur in neither the users' turf nor the software professionals' turf). The practices in between the extremes are hybrid practices, and constitute the third space of PD. As we explore hybrid methods that occur in this third space, we can look for HCI analogies of the attributes and advantages that are listed for third space studies in Table 49.1.

49.3 THIRD SPACE: NEGOTIATION, SHARED CONSTRUCTION, AND COLLECTIVE DISCOVERY IN PARTICIPATORY DESIGN AND HUMAN–COMPUTER INTERACTION

In the remaining sections of the chapter, we will describe a diversity of PD techniques, methods, and practices that provide hybrid experiences or that operate in intermediate, third spaces in HCI. Because our theme is hybridity, we have organized these descriptions in terms of strategies and moves that introduce novelty, ambiguity, and renewed awareness of possibilities, occurring at the margins of existing fields or disciplines (see Table 49.1). In several cases, a single report may fall into several categories. For example, Ehn and Sjögren (1991) conducted a workshop (see Section 49.4.2) in which a storytelling method (see Section 49.5.1) provided a space in which people negotiated the naming and defining of workplace activities (see 49.7.1). We hope that the strategies and moves of the PD practitioners and researchers will become clear, despite the multiple views onto individual reports.

49.4 SPACES AND PLACES

49.4.1 Sitings

One of the simplest parameters that can be manipulated to influence hybridity is the site of the work. At first, this appears to be a simple issue. As Robins (1999) says, "There are two approaches to participatory design: (1) Bring the designers to the workplace and (2) Bring the workers to the design room." This binary choice reflects the taxonomic distinctions that we reviewed above. However, even within the binary choice, the selection of the site can be important. Fowles (2000), in a discussion of participatory architectural practice, provides an insight that can apply as well for HCI: "If possible[,] design workshops should be located in the locality of the participating group and in the School of Architecture. Bringing the public into the School helps to de-mystify the profession, and taking students in the community furthers their understanding of the problem and its context" (p. 65). Pedersen and Buur (2000), in their work on industrial sites, agree (italics in the original):

> When collaborating with users *in our design environment* (e.g., a meeting space at the company), we can invite a number of users from different plants and learn from hearing them exchange work experiences.… Being in a foreign environment (and with other users), users will tend to take a more general view of things.
>
> When collaborating with users *in their work context*, users tend to feel more at ease as they are on their home ground—we are the visitors. Tools and environment are physically present and easy to refer to. This makes for a conversation grounded in concrete and specific work experiences.
>
> The idea was born to create a type of design event with activities in both environments and with two sets of resources to support design collaboration.

In a study of telephone operators' work conducted by one of us, we held our sessions at operator service offices and in research offices (Muller et al. 1995a). The work site meetings had the advantages of easy access to equipment on which we could demonstrate or experiment. During those meetings, there was a sense of being strongly tied to practice. The research site meetings were less tied to specific practices, and had a tendency to lead to more innovative ideas. Perhaps more subtly, the two different sites enfranchised different marginal participants. At the work site, it was easy to bring in additional work-domain experts (mostly trainers and procedures experts): They became adjunct members of the core analysis team for the duration of those meetings, *and* they became resources for the core team afterwards. At the research site, it was easy to bring in more technology experts, as well as the graduate students who later performed data analysis. The research site meetings became an occasion of enfranchisement, contribution, and early commitment for these additional actors. Both core and adjunct members from both sites became coauthors of our report (Muller et al. 1995a).

Brandt and Grunnet (2000) also considered site selection in their Smart Tool and Dynabook projects, which were concerned with working conditions in the office and in the home, respectively. In the Smart Tool case, they conducted dramatic scenarios in the project designers' environment. In the Dynabook case, they asked people at home to create and enact scenarios in their own living areas.

When University of Maryland researchers codesign with children, neither the school environment nor a traditional computer science lab is regularly used for ongoing PD projects that range from developing new digital libraries for children (Druin 2005) to creating new mobile storytelling devices (Fails, Druin, and Guha 2010). An after-school program that takes place twice a week during the school year and two weeks during the summer occurs in a lab that is specially carpeted for extensive use of the floor for designing. There are special windows that enable doors to be shut without concerns for safety or privacy. Although it is a lab that sits on a college campus, it is a third space where children and researchers can work together in a hybrid setting.

Brereton (2009; Segalowitz and Brereton 2009) takes an even stronger position, which combines traditional ethnography with action research. In her embedded research paradigm, the researcher lives as a member of the users' community for an extended period of time.

In addition, we note a related trend in community-based participatory research (CRPR), in which it is assumed that community members hold key knowledge and discernment about local needs, and that they can use this knowledge to help to solve both local and regional problems (Shallwani and Mohammed 2007). This approach has been used to frame technology and planning explorations for community needs (e.g., Corburn 2003; Shilton et al. (2008). Füller et al. (2006) used a variant of this idea which they called "community-based innovation" (CBI) to community sourcing of design ideas and design critiques from end users in an automotive design case.

Third space. In terms of hybridity, the selection of site can be a deliberate strategy to introduce new experiences and perspectives to one or more parties in the design process—a decentering move that can bring people into positions of ambiguity, renegotiation of assumptions, and increased exposure to heterogeneity. Returning to Bhabha's original argument, site selection initially appears to be a matter of *moving across the boundary* between different work cultures, rather than *living within the boundary*. However, the use of *common design practices across sites* makes those practices (and the membership of the design group) into a kind of movable third space. The practices and the group membership become stable features that persist across multiple sites. At the same time, the practices, and even the membership, grow and evolve with exposure to new sites and new understandings. In these ways, the practices become an evolutionary embodiment of the knowledge of the learning of the group (e.g., Floyd 1987; Muller 1997a).

Claimed benefits. What have practitioners gained through site selection, within this deliberately hybrid-oriented work area? Several themes emerge:

- *Improved learning and understanding.* Fowles (2000) described a move from a "symmetry of ignorance" toward a "symmetry of knowledge" as diverse parties educated one another through a "symmetry of learning"—and even a kind of "transformation" through exposure to new ideas (see also Carmien et al. [2003]). Brandt and Grunnet (2000), Pedersen and Buur (2000), Druin (2005), and Muller et al. (1995b) also claimed that the selection of site led to the strengthening of the voices that were comfortable at each site.
- *Greater ownership.* Petersen and Buur (2000) noted that their procedures strengthened user involvement in their project. Fowles (2000) and Muller (1995b; see also Muller, Wildman, and White [1994]) made specific reference to increases in commitment and ownership of the evolving knowledge and design of the group.

49.4.2 Workshops

Workshops may serve as another alternative to the two "standard" sites that most of us think about. In PD, workshops are usually held to help diverse parties ("interested parties" or "stakeholders") communicate and commit to shared goals, strategies, and outcomes (e.g., analyses, designs, and evaluations, as well as workplace-change objectives). Workshops are often held at sites that are in a sense neutral—they are not part of the software professionals' workplace and they are not part of the workers' workplace.

More importantly, workshops usually introduce novel procedures that are not part of conventional working practices. These novel procedures take people outside of their familiar knowledges and activities, and must be negotiated and collectively defined by the participants. Workshops are thus a kind of hybrid or third space, in which diverse parties communicate in a mutuality of unfamiliarity and must create shared knowledges and even the procedures for developing those shared knowledges.

The best-known workshop format in PD is the Future Workshop (e.g., Kensing and Madsen 1991; see also Bødker, Kensing, and Simonsen [2004]; McPhail et al. [1998]; Mørch, Engen, and Åsand [2004]), based on German civic planning (Jungk and Mullert 1987), a Future Workshop proceeds through three stages: (1) *Critiquing* the present; (2) *Envisioning* the future; and (3) *Implementing*—moving from the present to the future. These three activities involve participants in new perspectives on their work and help to develop new concepts and new initiatives.

A number of workshops have focused on simple materials and informal diagrams, rather than on formal notations. Bødker, Kensing, and Simonsen (2004) noted that, "The tools are simple diagrams or drawings with no special formalisms… because staff members participating in the workshops, as well as those to whom the results are later presented, typically have no experience with technical descriptions using [Information Technology]-originated formalisms" (p. 252).

Sanders (2000, 2006) described a family of "generative tools," activities that are selectively combined into strategic design workshops, under an overall conceptual "say-do-make" strategy that combines market research ("what people say"), ethnography ("what people do"), and PD ("what people make"). Activities include the construction of collages focused on thinking (e.g., "how do you expect your work to change in the future?"), mapping (e.g., laying out an envisioned work area on paper), feeling ("use pictures and words to show a health-related experience in your past"), and storytelling (see Sections 49.5.1 and 49.7.2). Dandavate, Steiner, and William (2000) and Vaajakallio and Mattelmäki (2007) provided case studies of Sanders' method.

Sanders' *say-do-make* framework can also be used, in an analytic decomposition, to describe participatory opportunities in more challenging design settings. Of course, ethnography is a prime example of the "see" strategy (see Chapter 45 in this book). O'Connor et al. (2006) explored a case in which, in effect, the "*do*" aspect of Sanders' method was the only means of communication for a codesigner with severe physical and speech disabilities. Cohene, Baecker, and Marziali (2005) explored some aspects of the "*make*" strategy in work with a codesigner who had Alzhiemer's disease, and her family and caregivers. Although neither of these papers was written with Sanders' analysis in mind, the framework provided by Sanders helps us to understand the range of possibilities and the creative responses of the researchers to codesigning under constrained circumstances.

In a different setting, Buur, Binder, and Brandt (2000) developed a workshop in which workers carried a mock-up of a proposed new device (see Section 49.7.1) through an industrial plant, recording how it would be used. They then acted out a 5-minute video scenario (see Section 49.5.1.2), which they subsequently presented to other similar worker teams in a workshop. Hultcrantz and Ibrahim (2002) used a similar method to concretize workshops similar to focus groups that were held with family members in their own homes. Pedell (2004) described a lower tech story-boarding workshop format in which people created narratives using photographs, putting them in sequences and altering in many cases (typically through the addition of speech bubbles to show what people were thinking or doing). Monk and Howard (1998) used a similar method, with less emphasis on photographs, to develop a "rich picture" of a work domain.

A novel workshop solution was needed when bringing older adults together with children, ages 7–11 (Xie et al. in press)—two historically underrepresented constituencies in the design of new technologies. A community center facility was used for its familiarity and availability to the children and the elders. Because of the diversity of participants, we faced challenges of both putting the two groups at ease and developing design methods that could accommodate active children and less-active adults. Earlier work had hybridized the design session by encouraging children to treat the entire floor as a design area (Druin et al. 2009). This was no longer possible if the children wanted their elderly design partners to engage in the design experience. Instead, this 2-day workshop began with "getting to know you" experiences, followed by "low-tech prototyping," a technique widely used in PD with adults (see Section 49.7.3). Once this blue-sky brainstorming was completed, separate discussions with stakeholders nurtured ideas to be further refined. This age-bridging work provides an example of suiting a workshop setting and dynamics to the needs of diverse participants. Cameron (1998), too, faced a different setting and problem and chose a workshop solution. This project dealt with safety issues in urban design in Baltimore and—like the METRAC program in Toronto (Nisonen 1994; see also Önder and Der [2007])—invited community members to contribute their domain expertise as people who lived with safety issues on an everyday basis. Cameron provided a manual, based on a professionally developed set of safety guidelines. Community members became community organizers, bringing the project topic and the proposed guidelines to their own constituencies. Two additional workshops refined the safety audit information from the constituencies, selected priority issues to fix, and adopted an action plan. Cameron observed that,

> One of the successful aspects of the Design for Safety workshop is that it provided a forum for a diverse group of people to productively discuss common problems and work through shared solutions and consensus. The workshops also showed that crime and safety were not solely the responsibility of the police, but that public works employees, traffic engineers, and especially residents must work together to envision as well as carry out the plan… Requiring that residents share the workshop information at community association meetings further assisted the transfer of responsibility from the workshop into the neighborhood.
>
> Related work is being done in the area of CRPR (e.g., (Shallwani and Mohammed 2007; Shilton et al. 2008), as discussed above.

Several other groups have developed repertoires of multiple workshops, from which they can select the type of workshop that is needed for a particular situation, site, or problem. Svanæs and Seland (2004) described six workshops; I list four formats that they considered successful here:

- Workshop 1. Theater, modeling clay, "design by accident," and improvisation with teenagers to explore "our mobile future"
- Workshop 2. Theater, brainstorming, and improvisation with a much more structured set of props (no modeling clay) for a different telecommunications project
- Workshops 4 and 5. Theater with audience-critique of performance (similar to Boal's Theatre of the Oppressed, described below), sometimes using structured props as well as "designing on the spot" for new concepts, for a hospital communication project
- Workshop 6. Videotaped field data as a point of common reference before theatrical work similar to workshops 4 and 5

Bødker, Kensing, and Simonsen (2004) described a repertoire of workshops. One subset of workshops was differentiated largely in terms of the artifact that was cocreated by the participants, such as freehand drawing (see also Monk and Howard [1998]), collages (see also Pedell [2004]; Sanders [2000]), affinity diagrams (see also Beyer and Holtzblatt [1998]), and timelines. Dray (1992) also used freehand drawing technique, but in a round-robin brainstorming "Braindraw" format in which *n* participants collaboratively drew *n* drawings, rotating the drawings throughout the group so that each drawing contained ideas created by each of the members of the group.

Less familiar artifacts were also used to define and differentiate workshops in the Bødker et al. survey. "Dead Sea Scrolls" are textual descriptions of the history of a business process. "Roll lists" are brief textual descriptions of all of the interested parties related to a business activity or a technology artifact. "Mapping" (also called "mind mapping"—see e.g., Buzan and Buzan [1996], for nonworkshop use of this technique) is the description of a problem area, business process, function, or other matter of interest in terms of a number of briefly stated concepts, connected by lines or arcs. A special version of mapping constructs a "communication map" among persons or roles. Finally, "Prompted Reflections" can be used similarly to Dray's Braindraw technique (Dray 1992), to bring people with different design concepts into communication with one another.

In the domain of DPD, researchers have adapted old and new web technologies to support hybrid workshop-like activities. Heß, Offenberg, and Volkmar (2008) reported on the use of community servers to work with two configurations of end users—the "parliament community" and the "central committee" community. These two user forums provided guidance on the development of multimedia software for linking televisions and computers. Costabile and colleagues have developed a set of virtual workshops called "Software Shaping Workshops," in which medical staff from diverse roles in a hospital can collaborate with software technologists in design of tailored user interfaces that meet the work needs of each role or discipline of the hospital staff (Costabile et al. 2006, 2007).

Third space. The various workshop approaches have several commonalities. Each workshop brings together diverse participants to do common work, to produce common outcomes (especially Bødker, Kensing, and Simonsen 2004), and to develop a plan of joint action (especially Kensing and Madsen 1991; Bødker, Kensing, and Simonsen 2004; McPhail et al. 1998; Mørch, Engen, and Åsand 2004). They are thus opportunities that require mutual education, negotiation, creation of understanding, and development of shared commitments. Each workshop takes place in an atmosphere and (often) in a site that is not "native" to any of the participants. Thus, all of the participants are at a disadvantage of being outside of their own familiar settings, and they must work together to define their new circumstances and relationships. The combination of diverse voices leads to syntheses of perspectives and knowledges.

Claimed benefits. Advantages claimed for these experiences in hybridity include the following:

- *Development of new concepts* that have direct, practical value for product design (Dandavate, Steiner, and William 2000; Kensing and Madsen 1991; Sanders 2000) or for community action (Cameron 1998)
- *Engagement* of the interested parties ("stakeholders") in the process and outcome of the workshop (Xie et al. in press)
- *Combinations of different people's ideas* into unified concepts
- *Production of artifacts* that are the expected and useful "inputs" to the next stage of the development process (Bødker, Kensing, and Simonsen 2004; Svanæs and Seland 2004; Xie et al. in press)

49.5 NARRATIVE STRUCTURES

49.5.1 Stories

Stories and storytelling have played a major role in ethnographic work since before there was a field called "HCI" (for review, see Crabtree [1998]; Suchman and Trigg [1991]; see also Chapter 45 in this book). Stories have also had an important history in HCI (see Carroll [1995]; Erickson [1996]; Muller [1999a]; see also Chapter 48 in this book). We will not attempt to review these areas. Rather, we will focus on those aspects of story-collecting and storytelling that involve the construction of third spaces and hybridity.

Stories in participatory work may function in at least four ways.* First, they may be used as triggers for conversation, analysis, or feedback (Salvador and Howells 1998; Salvador and Sato 1998, 1999). Second, they may be told by end users as part of their contribution to the knowledges required for understanding product or service opportunities and for specifying what products or services should do (Brandt and Grunnet 2000; Lafreniére 1996; Muller 2001; Muller et al. 1995b; Noble and Robinson 2000; Patton 2000; Sanders 2000; Tschudy, Dykstra-Erickson, and Holloway 1996; Yu and Liu 2006). Third, they may be used by heterogeneous design teams (i.e., including users) to present their concept of what a designed service or product will do, how it will be used, and what changes will occur as a result (Demirbilek and Demirkan 2004; Druin 1999; Druin et al. 2000; Ehn and Kyng 1991; Ehn and Sjögren 1986, 1991; Gruen 2001; Muller, Wildman, and White 1994; Sanders 2000). Fourth, they may be constructed by designers to stand as proxies for real users (e.g., Triantafyllakos, Palaigeorgiou, and Tsoukalas 2010).

Beeson and Miskelly (1998, 2000) used hypermedia technologies to enable communities to tell their own stories, with the intention that "plurality, dissent, and moral space can be preserved" (Beeson and Miskelly 2000, p. 1). They were concerned to allow multiple authors to reuse community materials selectively, telling different stories within a common context. The different accounts were organized according to themes and laid out spatially on the image of a fictitious island for navigation by end users. Their work entered several areas or aspects of hybridity. First, the authors of the stories (i.e., community members) were using hypermedia technology for the first time, and were thus in the role of learners, even while they were the owners of the stories, and were thus in the role of experts. Second, the authors wrote from their own perspectives, which were sometimes in strong conflict with one another. Third, the authors could make use of one another's materials, effectively moving away from single-author narratives and into a kind of collaborative collage of materials, which conveyed interlinked stories. Fourth, just as the community members were negotiating and defining their roles as learner-experts, the software professionals/researchers were negotiating and defining their roles as experts/facilitators/students. Törpel and Poschen (2002) described a related

* For a survey of story genres that may be used in participatory work, see Karasti, Baker, and Bowker (2002).

method of Narrative Transformation, emphasizing workers' roles as story creators, story analysts, and originators of new concepts that could be pursued through other methods in this chapter (e.g., see Section 49.7.3).

A second line of practice and research has emphasized end users telling their stories using a system of paper-and-pencil, card-like templates. The earliest version was the collaborative analysis of requirements and design (CARD) technique of Tudor et al. (1993), later developed into a more general tool in Muller et al. (1995b) and further refined in Muller (2001). Lafreniére (1996) developed a related practice, collaborative users' task analysis (CUTA), repairing some of the deficits of CARD for his settings. Halskov and Dalsgård (2006) focused the method on combinations of "domain cards" with "technology cards" (see also Davis [2010]). Tschudy, Dykstra-Erickson, and Holloway (1996) developed their own highly visual version, PictureCARD, for a setting in which they had no language in common with the users whose stories they wished to understand. The card-based practices used pieces of cardboard about the size of playing cards. Each card represented a component of the user's work or life activities, including user interface events (i.e., screen shots), social events (conversations and meetings) and cognitive, motivational, and affective events (e.g., the application of skill, the formation of goals or strategies, surprises and breakdowns, and evaluations of work practices). The cards were used by diverse teams in analysis, design, and evaluation of work and technology. Because the cards were novel object to all the participants, they occasioned third-space questionings and negotiations, resulting in new shared understandings and coconstructions. Often, teams used the cards to prepare a kind of story-board poster, narrating the flow of work and technology use and annotating or innovating cards to describe that work. The resulting posters formed narratives of the work that were demonstrated to be understandable to end users, corporate officers, and software professionals, and which led to insights and decisions of large commercial value (see Sanders [2000], for a differently constructed example of story-board posters to describe work).

Druin (1999; Druin et al. 2000) pursued a third line of storytelling research and practice, with children as design partners in a team that also included computer scientists, graphic designers, and psychologists (for other participatory work with children, see e.g., Sanders [2000]; Hornof [2008]; Kam et al. [2006]; Large et al. [2007]; Taxén [2004]). Their purpose was to envision new technologies and practices in children's use of computers and related devices. They used both online story-boarding techniques and the construction of prototypes of spaces in which the jointly authored stories could be performed. This work kept everyone learning from everyone else—children learning about technologies and the story-boarding environment, adults learning about children's views and other adults' expertises, and everyone negotiating the meaning of new technological and narrative ideas, as well as their implementations.

So far, this section has addressed primarily the acquisition of stories. But stories are also for telling to others. Sanders (2000) described the construction of storyboards based on users' experiences. Gruen (2000, 2001) described guidelines and practices through which a diverse team could begin with a concept, and then could craft a convincing and engaging story around it. Demirkbilek and Demirkan (2004) used stories initiated by seniors in Turkey to redesign household items for greater usability by elder people. Massimi and Baecker (2006) similarly used seniors' stories for the redesign of mobile telephones.

Triantafyllakos, Palaigeorgiou, and Tsoukalas (2010) described a method for creating rich characters around which designers could consider design alternatives—an approach similar to the "personas" approach of Cooper, Reimann, and Cronin (2007).* Best et al. (2009) present a contrasting case, in which members of a diaspora community (i.e., citizens living outside of their own country) served as a proxy for their less well-traveled citizens at home, with results that in some ways showed the weakness of using proxies for actual users. Going further in the direction of contextualized knowledge, Brereton advocates for a participatory approach that she called "embedded research," in which the researcher lives as a member of the users' community for an extended period of time (2009; Segalowitz and Brereton 2009). In general, the problem of "designing for the 'other'" (Nielsen and Bødker 2009; see also Hirsch [2009]) remains an open question in PD, as in all of user-centered design (Stappers et al. 2009). That is, how can people speak for themselves if they are not even present? How can designer verify their knowledge of the users if the users are not available to discuss their needs?

Sanders' and Gruen's procedures led to hybrid experiences, in the sense that few software professionals or end users think in terms of story-construction or rubrics for effective fictions. Irestig and Timpka (2002) described a method for sharing stories from small working groups with a larger audience of decision makers.

Third space. Story-collecting and storytelling generally require a kind of third space in which to occur. Beeson and Miskelly (1998, 2000) were specifically concerned to create a new space for story-writing and story-reading and to maintain some of the most important aspects of third spaces in that new space—that is, preservation and expression of new meanings, relationships, conflicts, multiple perspectives, and "heterotopia." The three card-based practices use unfamiliar media (the cards), and made those media central to the team's activities, thus requiring conscious attention to shared conceptualizing and defining of those media, as well as the creation of new media when needed. Druin and colleagues created new software environments and new devices to craft and implement stories of futuristic technologies. Finally, Gruen engaged diverse teams in new roles as story writers, guided by

* See also critiques of the personas approach such as in Adlin et al. (2006).

expert-derived guidelines, in the writing of professionally structured and professionally paced stories for organizational or commercial use.

Claimed benefits. The story-collecting and storytelling practices are diverse, and serve multiple purposes. A brief summary of the claims of their value to projects and products is as follows:

- *Articulation* and preservation of a diverse community's views (Beeson and Miskelly 1998, 2000)
- *Practical application* to work analysis, task analysis, new technology innovation, and usability evaluation in commercially important products and services (Gruen 2000, 2001; Lafreniére 1996; Muller 2001; Muller et al. 1995b; Sanders 2000; Tudor et al. 1993; Tschudy, Dukstra-Erickson, and Holloway 1996)
- *Cocreation of new ideas* and children's articulation and self-advocacy (Druin 1999; Druin et al. 2000)

49.5.1.1 Photographs

There are many ways to tell stories. One approach that has informed recent PD work is end-user photography. Patton (2000) note that both (1) taking pictures and (2) organizing pictures into albums are, of course, familiar activities to most people in affluent countries. These activities allow end users to enter into a kind of native ethnography, documenting their own lives. In keeping with the issues raised in Section 49.5.1, it is important that the informants themselves (the end users) control both the camera and the selection of images (see Bolton [1989], for a set of discussions of the uses and abuses of documentary photography). They thus become both authors and subjects of photographic accounts of their activities. This dual role leads to one kind of hybridity, in which the photographic activities partake of both the world of common social life and the world of documenting and reporting on working conditions.

To address the problem that "rural women are often neither seen nor heard," Wang, Burris, and Ping (1996) in collaboration with the Yunnan Women's Health and Development Program, invited Chinese village women to articulate their lives through *photo novellas* created with cameras that the women controlled, with the goal of influencing policy makers. In an exploration of products for mobile knowledge workers, Dandavate, Steiner, and William (2000) similarly asked their informants to take pictures as part of a documentation of the working lives. In their study, informants were also invited to construct collages of their working lives, selectively reusing the photographs (among other graphical items) in those collages. The collages were, in effect, one type of interpretation by the photographers of their own photographs. Similarly to Patton's work, Dandavate et al. asked their informants to go out of their conventional professional roles as office workers (but well within their roles as members of an affluent culture) in the activity of taking the photographs. Dandavate et al. asked their informants to go even further out of role, through the construction of the collages based on their photographs and the interpretation of the collages. The activities were thus marginal, partaking of attributes of informal life and professional life, of familiar and unfamiliar activities. They concluded that the photographic work led to new learnings and understandings that had not been accessible through observational studies, as well as a stronger sense of ownership by their informants in the outcome of the study.

Noble and Robinson (2000) formed an alliance between an undergraduate design class at Massey University and a union of low-status service workers, developing photo documentaries of service work. The photographs served as a kind of hybrid boundary object (Star and Griesemer 1989)—for the students, the photographs were composed artifacts of design, whereas for the union members, the photographs were common and casually produced snapshots. Discussions between union members and students were rich, conflicted, and productive, as they negotiated the status and meaning of these hybrid objects. These discussions—and the exhibits and posters that they produced (i.e., the collective actions of the students and the union members)—could not have been successful without mutual learning and construction of new understandings. Photo documentaries were used by Kwok (2004) as a means of providing familiar, concrete artifacts to enable design collaborations. Mattelmäki and Batarbee (2002; see also Hulkko et al. [2004]) used photo documentaries as one component of a set of user-composed diary techniques, with a subsequent user-created collages to serve as a rich source of discussions.* Taylor and Cheyerst (2009) further pursued themes of lay photography and group reflection through a community-scaled photo display device.

Pecknold (2009) developed a novel mixture of photography, drawing, and "probes" in order to conduct remote design dialogues between her university in Canada and her informants in Rwanda. Women answered a prepared set of questions through photographs and drawings and labeled self-selected photos and drawings to correspond to further questions about hopes and desires. Like the tailoring of the workshop setting for elders and children (see Section 49.4.2), this is another example of suiting a previously well-understood set of participatory methods to the special circumstances and special needs of a new group of participants.

Third space. End-user photography is an interesting case of hybridity and the production of third spaces. Photography is a good example of an "in-between" medium—one that is part of many people's informal lives (Dandavate, Steiner, and William 2000; Noble and Robinson 2000; Patton 2000), but that is also an intensively studied medium of communication and argumentation (Bolton 1989; Noble and Robinson 2000). Photography occurs at the margin

* It is noteworthy that, in the studies reviewed here, the informants made their own decisions about what was important, and therefore what they should photograph. For a discussion of issues in more conventional, researcher-directed photographic diary studies, see Carter and Mankoff (2005).

of most people's work, and yet can easily be incorporated into their work. The resulting photographs and drawings in these projects have attributes of their dual worlds—they are partially informal and quotidian and partially formal and documentary. Discussions around the photographs and combination of the photographs into photo narratives (Kwok 2004; Patton 2000) or collages (Dandavate, Steiner, and William 2000; Hulkko et al. 2004; Mattelmäki and Batarbee 2002) can lead to mutual learning and new ideas, particularly through the inclusion of the voices of the photographers, the viewers, and especially the people depicted in the photographs (Noble and Robinson 2000; see also discussion of Isomursu, Kuutti, and Väinämö [2004], below). Because photographs are often thought of as denotative media (i.e., documenting what *is*), Pecknold's approach of supplementing photographs with more connotative drawings is very promising for helping people to express and communicate their hopes and desires about possible futures (Pecknold 2009).

Claimed benefits. The use of end-user photographs and drawings appears to be new and experimental, and there are few strongly supported claims of benefits. Informal claims of success and contribution include the following:

- *Richer, contextualized communication medium* between end users and designers (in some cases, the designers were not, themselves, software professionals)
- *Stronger engagement* of designers with end-users worlds
- *Enhanced sharing* of views and needs among end users, leading to stronger articulation by them as a collective voice
- *Expression* of emotions and other connotative concepts, as well as documentation of more denotative, fact-like information

49.5.1.2 Dramas and Videos

Drama provides another way to tell stories—in the form of theatre or of video. One of the important tensions with regard to drama in PD is the question of whether the drama is considered a finished piece, or a changeable work in progress.

Many PD drama practitioners make reference to Boal's Theatre of the Oppressed (Boal 1974/1992). Boal described theatrical techniques whose purpose was explicitly to help a group or a community find its voice(s) and articulate its position(s). The most influential of Boal's ideas was his Forum Theatre, in which a group of nonprofessional actors performs a skit in front of an audience of interested parties. The outcome of the skit is consistent with current events and trends—often to the dissatisfaction of the audience. The audience is then invited to become authors and directors of the drama, changing it until they approve of the outcome. A second technique of interest involves the staging of a tableau (or a "frozen image," in Brandt and Grunnet 2000), in which a group of nonprofessional actors positions its members as if they had been stopped in the middle of a play. Each member can tell what she/he is doing, thinking, planning, and hoping. Forum Theatre was used informally in the UTOPIA project and other early Scandinavian research efforts (Ehn and Kyng 1991; Ehn and Sjögren 1991), addressing the question of new technologies in newspaper production. Changes in work patterns and work-group relations were acted out by software professionals in the end-users workplace, using cardboard and plywood prototypes, in anticipation of new technologies. The workers served as the audience and critiqued the envisioned work activities and working arrangements. The drama was carried out iteratively, with changes, until it was more supportive of the skilled work of the people in the affected job titles. The researchers made repeated visits with more detailed prototypes, again using the vehicle of a changeable drama, to continue the design dialogue with the workers. This work was widely credited with protecting skilled work from inappropriate automation, leading to a product that increased productivity while taking full advantage of workers' skills. Brandt and Grunnet (2000) made a more formal use of Boal's Forum Theatre and "frozen images" in the two projects described above (Section 49.4.1). Working with refrigeration technicians in the "Smart Tool" project, they and the technicians enacted work dramas and tableaux around four fictitious workers, leading to insights about the technicians' work and the technological possibilities for enhanced support of that work. Here is a description of one use of Forum Theatre:

> [T]he stage was constructed of cardboard boxes which in a stylized way served as… the different locations in the scenario. At first the service mechanics sat as an audience and watched the play. After the first showing of the "performance" the refrigeration technicians were asked to comment and discuss the dramatized scenario critically…
>
> The role of the refrigeration technicians changed from being a passive audience into being directors with an expert knowledge. The users recognized the situations shown in the dramatized scenario…. Because of the openness of the scenario there was a lot of "holes" to be filled out. For instance, one…technician explained that he preferred to solve the problems himself instead of calling his boss. This information meant that the Smart Tool should be able to help him solve his problems while being in his car.… Another [technician] wanted to have personal information that his boss was not allowed…[to] access…. (p. 14)

Incidents were analyzed through tableaux. The designers positioned themselves in the "frozen image" of the work situation, and then led a discussion of (1) the work activities that were captured in the stopped action, and (2) the work relations in which each particular tableau was embedded.

Muller et al. (1994) presented a related tutorial demonstration piece called "Interface Theatre," with the stated goal of engaging a very large number of interested parties in a review of requirements and designs—for example, in an auditorium. In Interface Theatre, software professionals acted out a user interface "look and feel" using a theatrical stage as the screen, with each actor playing the role of a concrete interface component (e.g., Kim the Cursor, Marty the Menubar, Dana the Dialoguebox). Pedersen and Buur (2000; see also Buur, Binder, and Brandt [2000]), following previous work of Binder (1999), collaborated with industrial workers to make videos showing proposed new work practices and technologies. After a collaborative analysis of the work (see Section 49.6), workers acted out their new ideas and took control of which action sequences were captured on video for subsequent explanation to other workers and management (see also Björgvinsson and Hillgren [2004]; Mørch, Engen, and Åsand [2004]). Isomursu, Kuutti, and Väinämö (2004) used more informal user-produced videos based on cellphone video recordings, which included not only lay-ethnographic records of usage, but also user-originated dramas to illustrate hypothesized or desired aspects of usage. In the situated and participative enactment of scenarios method, Iacucci et al. described a projective series of improvisations with an innovative technology idea—the "magic thing"—in users' homes or workplaces (Iacucci, Iacucci, and Kuutti 2002; Iacucci and Kuutti 2002; Kuutti, Iacucci, and Iacucci 2002; see also Buur and Bødker [2002]; Bødker and Buur [2002]).

Finally, Salvador and Sato (1998, 1999) used acted-out dramas as triggers for questions in a setting similar to a focus group, and Howard et al. (2002) described the role of professional actors and directors in dramatizing attributes of proposed new products. Kantola et al. (2007; Kankainen et al. 2005) similarly used dramatic readings by "role characters" to deepen the understanding of users' situations. Enquist and Tollmar (2008) used role-playing as part of a series of workshops to envision a future health-related memory aid for pregnant women.

While all of these practices are loosely tied together through the use of drama, there are important contrasts. One important dimension of difference is the extent to which the drama is improvised in the situation, or scripted in advance. Boal's techniques make a crucial use of improvisation by the user-audience, to change the action and outcome of the drama. This theme is most clearly seen in the work of Brandt and Grunnet (2000), Ehn and Sjögren (1986, 1991), and Muller, Wildman, and White (1994). At the opposite extreme are video documentaries, which of course are difficult to change in response to discussion and constructive insight.

Third space. Taken as a somewhat diverse participatory genre, the dramatic approaches provide many of the aspects of hybridity reviewed in the cultural studies introduction to this chapter. Drama brings a strong overlap of the world of end users and the world of technology developers/researchers, showing concrete projections of ideas from one world into the other world—and, in most uses, allowing modification of those ideas. Drama is marginal to the work domains of most technology developers/researchers and most end users, and thus moves all parties into an ambiguous area where they must negotiate meaning and collaboratively construct their understandings. Agreements, conflicts, and new ideas can emerge as their multiple voices and perspectives are articulated through this rich communication medium.

Claimed benefits. Similarly to end-user photography, most of the theatrical work has the feel of experimentation. It is difficult to find clear statements of advantages or benefits of these practices (see Section 49.8). In general, practitioners and researchers made the following claims:

- *Building bridges* between the worlds of software professionals and users
- *Enhancing communication* through the use of embodied (i.e., acted-out) experience and through contextualized narratives
- *Engaging small and large audiences* through direct or actor-mediated participation in shaping the drama (influencing the usage and design of the technology)
- *Increasing designers' empathy* for users and their work
- *Simulating use of not-yet-developed tools* and technologies ("dream tools," Brandt and Grunnet 2000) to explore new possibilities
- *Fuller understanding* by focus group members, leading to a more informed discussion

49.6 GAMES

From theory to practice, the concept of games has had an important influence in participatory methods and techniques. Ehn's theoretical work emphasized the negotiation of language games in the course of bringing diverse perspectives together in PD (Ehn 1988; for applications of this theory, see Ehn and Kyng [1991]; Ehn and Sjögren [1986, 1991]). In this view, part of the work of a heterogeneous group is to understand how to communicate with one another—and of course communication is not really possible on a strict *vocabulary* basis, but requires an understanding of the *perspectives* and *disciplinary cultures* behind the words (Bachmann-Medick 1996; Muller 1997a, 1997b, 1999b). Thus, the work of heterogeneous teams is, in part, the "mutual validation of diverse perspectives" that Bødker et al. (1988) advocated.

Games have also been an important concept in designing practices, with the convergent strategies of enhanced teamwork and democratic work practices within the team.* We explained the concepts as follows (Muller, Wildman, and White 1994):

> When properly chosen, games can serve as levelers, in at least two ways. First, games are generally outside of most workers' jobs and tasks. They are therefore less likely to appear to be "owned" by one worker, at the expense of the alienation

* For an example of games used to teach *design experiences* among students, see Iversen and Buur (2002).

> of the non-owners. Second,…[PD] games…are likely to be novel to most or all of the participants. Design group members are more likely to learn games at the same rate, without large differences in learning due to rank, authority, or background.… This in turn can lead to greater sharing of ideas.…
>
> In addition, games…can help groups of people to cohere together [and] communicate better. One of the purposes of games is enjoyment —of self and others —and this can both leaven a project and build commitment among project personnel. (pp. 62–63)

Derived from Ehn's (1988) theoretical foundation, Ehn and Sjögren (1986, 1991; see also Bødker, Grønbæk, and Kyng [1993]) adopted a "design-by-playing" approach, introducing several games into PD practice:

- *Carpentopoly,* a board game concerned with business issues in the carpentry industry.
- *Specification game*, a scenario-based game based on a set of "situation cards," each of which described a workplace situation. Players (members of the heterogeneous analysis/design team) took turns drawing a card and leading the discussion of the work situation described on the card. Hornecker (2010) used a more restricted approach, in which cards primarily asked questions about designed artifacts.
- *Layout kit*, a game of floor plans and equipment symbols, for a workers' view of how the shop floor should be redesigned (see also Bødker and Buur [2002]; Horgan et al. [1998]; Klær and Madsen [1995]; and most recently Brandt and Messeter [2004], reviewed below).
- *Organization kit and desktop publishing game*, a part of the UTOPIA project (Ehn and Kyng 1991), in which cards illustrating components of work or outcomes of work were placed on posters, with annotations.

Petersen and Buur (2000) extended the Layout Kit in new ways. Collaborating with workers at Danfoss, they jointly created a board game for laying out new technologies in an industrial plant:

> A map of the plant layout served as the game board.… Foam pieces in different colors and shapes worked as game pieces for the team to attach meaning to…. Often, in the beginning of the game, the placement of the piece was only accepted when touched by almost everybody…. The participants were forced to justify the placement, which fostered a fruitful dialogue about goals, intentions, benefits, and effects. People were asking each other such things as…"what if we change this?", "on our plant we do this, because…", "would you benefit from this?".

The games became the foundation of the videos produced in collaboration with the workers (described in Section 49.5.1.2).

Buur, Binder, and Brandt (2000) extended the Specification Game, making a game from the outcome of a participatory ethnographic analysis of work at an industrial plant. They first collected video observations from work activities and developed a set of 60–70 video excerpts for further discussion. They next constructed a set of cards, one for each video excerpt, with a still-frame image from the video displayed on each card. Same participants then grouped these 60–70 cards into thematic clusters, organized their clusters, and analyzed the subsets of actions in each cluster (for a related non-game technique, see affinity diagramming in Beyer and Holtzblatt [1998]). Similar approaches were used by de Jong, Kouprie and De Bruyne (2009; Bruyne and de Jong 2008) for self-reflection by workers on their behaviors in the context of the physical workplace, and to envision future possibilities (see also Maarleveld, Volker, and van der Voordt [2009]).

The concept of games was taken in a different direction, for use in non-Scandinavian workplaces, by introducing several new games (Muller, Wildman, and White 1994):

- *CARD*, a card game for laying out and/or critiquing an existing or proposed work/activity flow (see Section 49.5.1)
- *PICTIVE*, a paper-and-pencil game for detailed screen design (Muller et al. 1995b)
- *Icon design game*, a guessing game for innovating new ideas for icons (this game assumes subsequent refinement by a graphic designer)
- *Interface theatre*, for design reviews with very large groups of interested parties (see Section 49.5.1.2)

These games emphasized hands-on, highly conversational approaches to discussing both the user interface concept itself and the work processes that it was intended to support. We attempted to foster an informal and even playful tone, for the reasons sketched in the earlier quotation. Similar approaches have been used for design across barriers of disability (Davies et al. 2004) and across barriers of language and culture (Bidwell et al. 2010; Tschudy, Dykstra-Erickson, and Holloway 1996).

Recently Brandt and Messeter (2004; see also Johansson et al. [2002]) developed a strong sequence of games. Their User Game is based on the video-collage methods of Buur, Binder, and Brandt (2000), combining brief video clips into person or role descriptions, which are then labeled evocatively by the participants. The second game in their sequence, the Landscape Game, places those user constructs into the work environment (as a board game). The Technology Game adds simple shapes that stand for technologies, again playing those shapes onto the work environment in the Landscape Game. Finally, the Scenario Game moves back to the real world, enacting possibilities based on new ideas from the preceding three games. The enactments may be video recording, both for documentary purposes and to generate further video material for another cycle of the four games.

The goal of *designing a game* can also serve as an opportunity to create a hybrid space: The design task mixes aspects of software design and implementation with game-based concepts of enjoyment, suspense, and personal outcomes. Kam et al. used this strategy to engage students, their families, and their communities in workshops in a rural Indian village (Kam et al. 2006; see also Antle [2003]).

Third space. Each of these games took all of its players outside of their familiar disciplines and familiar working practices, but strategically reduced the anxiety and uncertainty of the situation by using the social scaffolding of games. Each game required its players to work together through mutual learning to understand and define the contents of the game and to interpret those contents to one another in terms of multiple perspectives and disciplines. The conventional authority of the technology developers/researchers was thus replaced with a shared interpretation based on contributions from multiple disciplines and perspectives.

Claimed benefits. PD work with games has been claimed to lead to the following benefits:

- *Enhanced communication* through the combination of diverse perspectives
- *Enhanced teamwork* through shared enjoyment of working in a game-like setting
- *Greater freedom to experiment and explore new ideas* through flexible rules and redefinition of rules during the game
- *Improved articulation* of the perspectives, knowledges, and requirements of workers
- *New insights* leading to important new analyses and designs with documented commercial value

49.7 CONSTRUCTIONS

Preceding sections have considered hybridity in participatory activities such as sitings, workshops, stories, photography, dramas, and games. This section continues the survey of participatory practices that bring users and technology developers/researchers into unfamiliar and ambiguous "third space" settings. In this section, we focus on the collaborative construction of various concrete artifacts:

- *Physical reflections of a cocreated language* of analysis and design
- *Descriptions of work* in unfamiliar media
- *Low-tech prototypes* for analysis and design
- *High-tech prototypes* for design and evaluation

49.7.1 Language

An earlier section noted Ehn's theoretical work on *PD as language games* (Ehn 1988). Ehn's interest converges with Bhabha's "third space" argument (Bhabha 1994): Part of the characterization of hybridity was the negotiation and cocreation of working language and meaning. This section takes Ehn's position seriously and considers the role of language creation in participatory practices that lead to hybridity.

Several projects have made physical objects into a kind of vocabulary for work analysis, design, or evaluation. The cards described in the Section 49.6 are examples (Buur, Binder, and Brandt 2000; Ehn and Sjögren 1986, 1991; Lafreniére 1996; Muller 2001; Muller et al. 1995b; Tschudy et al. 1994; Tudor et al. 1993). In each of these methods, the cards became a kind of "common language" (e.g., Muller et al. 1995b) through which the design team communicated (1) with one another, and (2) with their labor and management clients. In two of the methods, the cards themselves were acknowledged to be incomplete, and part of the work of the team was to develop and refine the cards so as to reflect their growing understanding and their new insights (Lafreniére 1996; Muller 2001). Team members (users and others) were encouraged to disregard, if appropriate, the template of information on each card, up to and including the decision to turn the card over and write on its blank back. In subsequent sessions, the concepts that were written on the blank backs of cards usually became new kinds of cards. The working vocabulary of the team thus grew as the shared understanding of the team grew. This extensibility of the set of cards was observed in nearly all sessions, but was particularly important in sessions that were envisioning future technologies or future work practices. The cards thus became a point of hybridity, where assumptions were questioned and challenged, where extensive and polyvocal dialogue was required for the team to assign meaning to the cards, where conflicts were revealed and resolved, and where the team had to construct its understanding and its language. Similarly, the board games of Ehn and Sjögren, and especially of Pedersen and Buur (2000), used deliberately ambiguous playing pieces. The analysis team had to assign meaning to the pieces, and did so in a collaborative way.

Chin et al. (2000), working with a community of physical scientists who were not software professionals, introduced software-like flowcharts to their clients (see Kensing and Munk-Madsen [1993], for a discussion of the relationship between concrete tools and abstract tools). This work shared, with the other work reviewed in this section, aspects of symbol ambiguity and language cocreation:

> To attune scientists to the construction of workflow diagrams, we provided them a simple, informal example of how a meteorologist might diagram his [sic] work in collecting and reporting weather conditions.… Although we used circles and arrows in our example, we did not impose any specific symbology or rules on the scientists' construction of workflow diagrams…. At times, the scientists did struggle in developing some diagrams, but the labor was mostly centered on the elucidation of the research processes rather than the mechanics of diagramming.

Third space. Common to all of these projects was the cocreation of a physically represented language, both within the team and from the team to its clients and stakeholders. This kind of lay linguistic work requires mutual education and mutual validation for the new language components to have meaning to all of the parties. These negotiations of multiple knowledges are at the heart of the "third space" proposal of Bhabha (1994).

Claimed benefits. Most of these projects involved a number of activities and a number of aspects of hybridity. It is difficult to determine how much of their successes were due specifically to the language-related components. Benefits that *may* have resulted from the negotiation and cocreation of language include the following:

- *Enhanced understandings* of one another's perspectives and needs
- *Critical examinations of assumptions* underlying the ways that each party expressed its perspective
- *Shared ownership of the language* and its physical manifestation (cards, flowcharts, and game pieces)
- *Improved communication* within the team and from the team to interested outsiders (clients and stakeholders)

49.7.2 Making Descriptive Artifacts

Another way of moving end users into unfamiliar and hence reflective experiences is to ask them to use "projective" or artistic methods to report on their experiences and needs. In one sense, these methods produce another kind of language of expression, and therefore might have been included in the preceding section. Because the outcomes are so distinctively different from the language-oriented work of the preceding section, we thought it best to review this work in its own section.

Sanders has employed user-created collage in her participatory practice for a number of years (Sanders 2000; see also Dandavate, Steiner, and William [2000]; Sanders and Branaghan [1998]; Sanders and Nutter [1994]). The choice of collage is of course strategic: Relatively few people make collages as part of their work activities and relatively few people interpret their collages to one another as part of their work conversations. Yet the content of the collages is strongly anchored in what people know. The collages thus become marginal constructions, not part of any defined workplace field or discipline, but informed by familiar knowledges. The novelty of the collage encourages the challenging of assumptions, and the interpretation and presentation of collages encourage mutual learning across the diversity of experiences and knowledges of the participants.

For completeness, we make reference to the work of Noble and Robinson (2000) on collaborative creation of photo documentaries and of Patton (2000) on end-user creation of photo collages, reviewed in Section 49.5.1.1 Their work also produced descriptive artifacts that took users and their collaborators into unfamiliar areas.

Third space. These methods have in common the use of a nonstandard medium for making users' needs known and for developing new insights in a workplace setting. The making of collages may be new for many participants. They are thus in a kind of "third space," between their work culture and the artistic or expressive culture of collages, and they have to reflect on the differences as they construct their approach to making collages of their own experiences. It is not clear, in Sanders' work, whether the collage work is done collaboratively among end users or whether each collage is a solitary production. If the collage creation is done collaboratively, then it might give rise to some of the other attributes of hybridity in Table 49.1—for example, challenging assumptions, cocreation of meanings and collective actions, and dialogues.

Claimed benefits. Based on her claims on years of practice with collages and related practices, Sanders (2000) claims the following benefits:

- *Using visual ways* of sensing, knowing, remembering, and expressing
- *Giving access and expression to emotional side* of experience
- *Acknowledging the subjective perspective* in people's experiences with technologies
- *Revealing unique personal histories* that contribute to the ways that people shape and respond to technologies

49.7.3 Low-Tech Prototypes

Beaudouin-Lafon and Mackay have provided a chapter on prototyping—including participatory prototyping—in this book. Therefore, we have written a very brief account in this chapter so as not to duplicate their efforts.

Low-tech prototypes may lead to "third space" experiences because they bring people into new relationships with technologies—relationships that are "new" in at least two important ways. First, the end users are often being asked to think about technologies or applications that they have not previously experienced. Second, in *participatory* work with low-tech prototypes, end users are being asked to use the low-tech materials to reshape the technologies—a "design-by-doing" approach (Bødker, Grønbæk, and Kyng 1993). In this way, participatory work with low-tech prototypes involves much more user contribution and user initiative than the more conventional use of "paper prototypes" as surrogates for working systems in usability testing (e.g., Rettig 1994). The general approach of low-tech prototyping for design has been effective in many settings, including with workers (Bødker et al. 1987, 1988; Bødker, Grønbæk, and Kyng 1993; Ehn and Kyng 1991; Lafreniére 1996; Muller 1991, 1992; Muller et al. 1995b); intercultural communication (Bidwell et al. 2010; see also Bidwell and Hardy [2009]) even when there is no common language (Tschudy, Dykstra-Erickson, and Holloway 1996); with people with disabilities (Moffatt et al. 2004); and with very young users (Druin 2002; Druin et al. 2009) and very old users (Massimi and Baecker 2006; Massimi, Baecker, and Wu 2007; see also literature reviews in Massimi [2006, 2007]).

The UTOPIA project provided impressive demonstrations of the power of low-tech cardboard and plywood prototypes to help a diverse group to think about new technologies, office layouts, and new working relations that might result from them (Bødker et al. 1987, 1988; Bødker, Grønbæk, and Kyng 1993; Ehn and Kyng 1991; for other use of low-tech, substitutive prototypes, see Mørch, Engen, and Åsand [2004]). Subsequent projects to translate this work to North America led to the PICTIVE method of paper-and-pencil constructions of user interface designs by heterogeneous design teams (Muller et al. 1995b); prototyping of consumer appliances using foam core and hook-and-loop attachments (Sanders and Nutter 1994); and a more experimental simulation of e-mail, using paper airplanes (Dykstra and Carasik 1991).

In addition to these methods, many researchers who work with children in PD experiences use low-tech prototyping. The children affectionately call it "bags of stuff" (Druin et al. 2009). The types of materials that are used are intentionally three-dimensional to cut down on the "fear of drawing" and to use these artifacts as a bridge for communication and design. Everything from toilet paper rolls to clay and cotton balls are used to construct new ideas with children and adults. These artifacts are then presented to a larger group and the highlights of the design ideas are written up on a whiteboard. The ideas are then aggregated to suggest a new design direction for the team (Druin 2002).

When prototyping takes place among geographically remote participants, the new situation is hybridized almost by definition Moore (2003) proposed an experimental approach to allow end users to create the appearance of the user interface and to provide rationales for their designs; it is not clear if this approach has been tested yet.* Rashid et al. (2006) took a critique-oriented approach to solve a similar participatory-requirements-analysis problem, providing a web method for users to create annotations with screen shots, which were then conveyed to the development team. Significantly, the Rashid et al. work was done during the design process, so that users were episodically involved in design critiques. Lohmann, Ziegler, and Heim (2008) described a text-plus-gesture method for critiquing designs through web browsers and conducted preliminary testing of the system with end users (for related work, see Lohmann et al. [2009]). Also addressing the problem of distributed requirements specification, Janneck and Gumm (2008) described the commented case studies method for collecting end-user information through scenario-based design-at-a-distance, sometimes involving a "Mediated Feedback" process to collect and redact user input (Gumm, Janneck, Finck 2006). Heß, Offenberg, and Volkmar (2008) described two online forum environments in which a "User Parliament" and a "Central Committee" of users and software professionals provided guidance for the duration of a community-driven development (CDD) process; see also the work of Füller et al. (2006), mentioned above, on CBI approaches to software design-at-a-distance.

Work in this newly defined area of DPD is in relatively early stages (Nasghsh et al. 2008). Many of the experiments involve repurposing of existing Web2.0 technologies to facilitate user feedback (e.g., We look forward to the maturity of this emerging effort.).

Third space. Low-tech prototyping has a reputation for bringing new insights through the combination of diverse perspectives. The UTOPIA project is widely credited with mutual education among shop-floor print workers and computer systems researchers. Experiences with PICTIVE and its variants almost always involved mutual education. Understanding and changing the artifact become important arenas for people to explore their understandings of one another's positions, to question one another's approaches, to discover and resolve conflicts, to engage in combinations of views leading to plans for collective action, and to accommodate heterogeneity of views and interests.

Claimed benefits. The low-tech participatory prototyping approaches have been extraordinarily influential, with adoption on four continents. Claimed benefits include the following:

- *Enhanced communication and understanding* through grounding discussions in concrete artifacts (Druin 2002)
- *Enhanced incorporation of new and emergent ideas* through the ability of participants to express their ideas directly via the low-tech materials and through the construction of artifacts that can be used in other techniques, especially drama and video documentaries (above)
- *Enhanced working relations* through a sense of shared ownership of the resulting design (Druin et al. 2009)
- *Practical application with measured successes* in using low-tech design approaches to real product challenges, achieving consequential business goals

49.7.4 Evolutionary Prototyping and Cooperative Prototyping

This last section on participatory methods is concerned with software prototyping. As noted above, we are relying on the chapter by Beaudouin-Lafon and Mackay in this volume to cover prototyping in greater depth and breadth. We include this brief overview for completeness of our chapter's survey of hybridity in participatory practices.

* One of us was involved in an earlier experiment called TELEPICTIVE, which attempted to support design-at-a-distance. We provided a description of the experimental prototype, and its shortcomings, in Miller, Muller, and Smith (1995) and Muller et al.

Bødker and Grønbæk (1991) and Madsen and Aiken (1993) explored the potential of cooperative prototyping in several projects, using different technology infrastructures. In general, they found that this approach led to enhanced communication with end users, improved incorporation of end user insights into the prototypes, and stronger collective ownership and collective action-planning by the team. They also observed time-consuming breakdowns in the design process itself, when new ideas required significant programming effort.

In a different prototyping approach, a system is delivered to its end users as series of iterative prototypes, each of which gradually adds functionality (e.g., Anderson and Crocca 1993; Bertelsen 1996; Trigg 2000). What appears to be critical is that the prototype functions as a *crucial artifact* in the end users' work—for example, a resource of documents for librarians (Anderson and Crocca 1993), an online event checklist that served as the crucial coordination point for the work of diverse contributions (Bertelson 1996), or a database supporting funding work in a nonprofit organization (Trigg 2000). Trigg (2000) provided a series of observations and tactical recommendations about how to engage the users in the evaluations that both they and the software professionals had agreed were needed.

In a rich survey of prototyping practices, Lim, Stolterman, and Teneberg (2008) took a different, more philosophically pragmatic approach to prototyping. In their analysis, prototyping has become a means for exploring a design space and for provoking questions within that space. Critical aspects of the prototype become the ability to *filter*, specifically to highlight the issues to be explored, while ignoring issues that could be distracting. The two case studies in Lim et al. involved conventional unidirectional prototyping—that is, *from* designer *to* user. Thus, these ideas have not yet been explored in a participatory context. It remains to be seen how these new ways of thinking about prototyping will affect participatory prototyping.

Third space. This very brief survey of cooperative prototyping and "iterative delivery" approaches shows several aspects of hybridity. In the case of cooperative prototyping, the cooperative work may be done in a physical third space that is neither the end-user's office nor the software developer's office (see Section 49.4.1). In the case of the delivery of iterated prototypes, each prototype is presented in the end-user's setting, but is unusual and only partially functional, and thus occasions reflection about its nature, its role in the end-user's work, and thus the work itself. In cases, the invitation (or perhaps the necessity) of the end-user's actions to help shape the technology becomes an important means of refocusing their attention, as well as the attention of the software developers. The ensuing conversations are concerned with the interlinked feasibility of changes to technology and to work practices, with attributes of hybridity including polyvocal dialogues, challenging one another's assumptions, and developing plans for collective actions.

Claimed benefits. Some of the virtues of the low-tech prototyping approaches have also been claimed for the cooperative prototyping and "iterative delivery" approaches:

- *Enhanced communication and understanding* through grounding discussions in concrete artifacts
- *Enhanced working relations* through a sense of shared ownership of the resulting design

Additional claims for software-based prototypes include the following:

- *Earlier understanding of constraints* posed by the practical limitations of software
- Improved contextual grounding of the design in the end-user's work practices

49.8 CONCLUSION

Our theme has been hybridity, and the ways in which selected methods in PD may bring useful attributes of hybridity or third space approaches into HCI work. We considered eight trends in PD—selection of sites of shared work, workshops, stories, end-user photography, dramas, creation of shared languages, descriptive artifacts (low-tech prototypes), and working prototypes—and we explored how each of these categories of practice may contribute to hybridity, and what advantages may result. The deliberate and selective use of hybridity has led to powerful methods in PD for increasing communication effectiveness, team coherence, innovation, and quality of outcome. Hybridity is thus at the heart of PD, fostering the critical discussions and reflections necessary to challenge assumptions and to create new knowledges, working practices, and technologies. When we consider HCI as a set of disciplines that lie between the space of work and the space of software development, we see that the hybrid third spaces developed within PD have much to offer HCI in general.

Table 49.2 summarizes the discussion of hybridity in PD, using the criteria derived from cultural studies (Table 49.1) and the experiences described in the eight areas of practice. Table 49.2 shows different patterns of hybridity for different methods, techniques, and practices.

Certain attributes are relatively common across practices—for example, inbetweenness, questioning assumptions, negotiation, and heterogeneity as the norm. Other attributes are relatively rare—for example, considerations of what constitutes legitimate data for analysis or design, how those data are analyzed as evidence, and how conclusions are drawn in each of the several fields that are represented in a team. These are difficult questions in the study of disciplinarity (Chandler, Davidson, and Harootunian 1994; Klein 1996), so it is perhaps not surprising that there is relatively weak support for

TABLE 49.2
Hybridity in Participatory Practices[a]

Attribute	Sitings	Workshops	Stories	Photos	Dramas	Games	Language	Descriptive	Prototypes
Overlap/Inbetweenness	?	+	−	+	+	+	+	+	+
Marginality	+	+	−	?	+	+	?	+	?
Novelty	+	+	?	?	+	+	+	+	+
Uncertain/shared “ownership”	?	+	?	−	+	+	+	−	−
Selected attributes	+	?	+	+	−	+	+	−	+
Conflicts	+	+	+	−	+	−	+	−	+
Questioning assumptions	+	?	+	+	+	+	+	?	+
Mutual learning	+	+	+	+	+	+	+	?	+
Synthesis of new ideas	?	+	+	+	+	+	?	+	+
Negotiation/(co)creation	+	+	+	+	+	+	+	+	+
Identities	−	−	+	+	−	?	?	+	?
Working language	−	?	+	+	−	+	+	+	+
Working assumptions and dynamics	+	?	+	+	+	+	+	?	+
Understandings	+	+	+	+	+	+	+	+	+
Relationships	?	+	+	+	−	+	?	+	?
Collective actions	?	+	?	+	?	?	?	+	+
Dialogues	+	+	+	+	+	+	+	+	+
Polyvocality	+	+	+	+	+	+	+	+	+
What is considered to be data?	−	−	−	+	−	−	+	+	−
What are the rules of evidence?	−	−	−	+	−	−	+	+	−
How are conclusions drawn?	−	−	−	?	−	−	+	−	−
↓ authority – ↑interpretation	+	?	+	+	+	+	+	?	+
↓ individualism – ↑collectivism	?	+	?	+	?	+	?	?	+
Heterogeneity as the norm	+	+	+	+	−	+	+	+	+

[a] Key: + practice includes this attribute of hybridity; − practice does not include this attribute; ? not sure

their exploration in participatory practices. For projects in which these are pivotal questions, we may need new methods that leverage hybridity in new ways. We hope that this survey of PD practices for creating third spaces will lead to new practices that strengthen these missing attributes. Conversely, I hope that new work in PD and HCI can help to ground discussions on some of the cultural studies in new ways.

This chapter would not be complete without a list of unsolved problems in PD:

- *Participation by nonorganized workforce.* The field of PD has long been concerned about how to engage in meaningful participative activities with workers or others who are not organized into a group with collective bargaining power or other collective representation (e.g., Greenbaum 1993, 1996; van den Besselaar, Greenbaum, and Mambrey 1996). This has been a particularly difficult problem when we have tried to compare methods from one country (and political culture) to another (e.g., Muller et al. 1991).
- *Evaluation and metrics.* One of the weaknesses of the literature on participatory practices is the dearth of formal evaluations. While there is general agreement that user involvement is beneficial in many aspects of analysis and design (e.g., Kujala 2003; see also Beyer and Holtzblatt [1998]; Cross [2001]; Dewulf and Van Meel [2002, 2003]; Garzotto [2008]; Pew and Mavor [2007]; Warr and O’Niell [2005]), the best way to structure and channel that “involvement” has been controversial (Druin 2002; Luck 2003; Olsson 2004; Reyman et al. 2005). There is a small set of papers that have examined software engineering projects across companies, and have found positive outcomes related to end-user participation (Cotton et al. 1988; Saarinen and Saaksjarvi 1990). We have been unable to discover any formal experiments comparing participatory methods with nonparticipatory methods in a credible workplace context. While it is possible to conduct design competitions in an academic environment (e.g., Peeters, von Tuijl, and Reyman 2008), the problems addressed are usually scaled to a classroom exercise, and the outcomes must be measured at a very early stage (e.g., *design* outcomes, not product outcomes). Indeed, such studies for real-world products and projects would be difficult to perform, because they would require that a product be implemented and marketed twice (once with participation and once without). The problem is made more difficult

because measurements and metrics of organizational outcomes, user participation, and user satisfaction are currently vexing research issues (e.g., Garrety and Badham 1998; Kappelman 1995; for review, see Gasson [1995]).

- *DPD*. It is already difficult to work across differences. Adding the problem of working across distances as well makes PD more difficult. In this chapter, we have reviewed work in DPD, and much of it is promising. We hope to see more specific methods and techniques that create new kinds of online spaces to continue this work.

REFERENCES*

Adlin, T., J. Pruitt, K. Goodwin, C. Hynes, K. McGrane, A. Rosenstein, and M. J. Muller. 2006. Putting personas to work. n *CHI 2006 Adjunct Proceedings*, 13–6. Montréal, QU, CA. ACM.

Anderson, W. L., and W. T. Crocca. 1993. Engineering practice and codevelopment of product prototypes. *Commun ACM* 36(6):49–56.

Antle, A. 2003. Case study: The design of CBC4Kids' StoryBuilder. In *Proceedings of IDC 2003*, 59–68. New York: ACM.

Anzaldúa, G. 1999. *La frontera/Borderlands*. San Francisco, CA: Aunt Lute Books.

Aune, M. 1996. The computer in everyday life: Patterns of domestication of a new technology. In *Making Technology Our Own: Domesticating Technology into Everyday Life*, ed. M. Lie and K. H. Sorensen, 91–120. Oslo: Scandivian University Press.

Bachmann-Medick, D. 1996. Cultural misunderstanding in translation: Multicultural coexistence and multicultural conceptions of world literature. *Erfurt Electronic Studies in English* 7. http://webdoc.gwdg.de/edoc/ia/eese/artic96/bachmann/7_96.html.

Badke-Schaub, P. 2004. Strategies of experts in engineering design: Between innovation and routine behavior. *Des Res* 4(2).

Balka, E. 1995. Political frameworks for system design: Participatory design in non-profit women's organizations in Canada and the United States. In *Workshop Proceedings: Political Frameworks of System Design From a Cross-Cultural Perspective*, ed. J. Gaertner and I. Wagner.

Beck, E. E. 1996. P for political? Some challenges to PD towards 2000. In *PDC'96 Proceedings of the Participatory Design Conference,* 117–25. Cambridge, MA: CPSR.

Beck, E. E. 2001. *On Participatory Design in Scandinavian Ecomputing Research*. University of Oslo. Oslo, Norway. Research Report 294.

Beeson, I., and C. Miskelly. 1998. Discovery and design in a community story. In *Proceedings of PDC 98,* 147–56. Seattle, WA: CPSR.

Beeson, I., and C. Miskelly. 2000. Dialogue and dissent in stories of community. In *Proceedings of PDC 2000,* 1–10. New York: CPSR.

Béguin, P. 2003. Design as a mutual learning process between users and designers. *Interact Comput* 15(5):709–30.

Bertelsen, O. W. 1996. The festival checklist: Design as the transformation of artifacts. In *Proceedings of PDC 96*, 93–102. Cambridge, MA: CPSR.

Best, M. L., T. N. Smyth, D. Serrano-Baquero, and J. Etherton. 2009. Designing for and with diaspora: A case study of work for the truth and reconciliation commission of liberia. In *CHI 2009 Adjunct Proceedings*, 2903–17. Boston, MA.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: Defining Customer-Centered Systems*. San Francisco, CA: Morgan Kaufmann.

Bhabha, H. K. 1994. *The Location of Culture*. London: Routledge.

Bidwell, N., and D. Hardy. 2009. Dilemmas in situating participation in rural ways of saying. In *Proceedings of OZCHI 2009*, 145–52. Melbourne, Australia. ACM.

Bidwell, N. J., T. Reitmaier, G. Marsden, and S. Hansen. 2010. Designing with mobile digital storytelling in Africa. In *Proceedings of CHI 2010*, 1593–602. Atlanta, GA. ACM.

Binder, T. 1999. Setting the scene for improvised video scenarios. In *Adjunct Proceedings of CHI 99*. Pittsburgh, PA: ACM.

Bjerknes, G., and T. Bratteteig. 1987. Florence in wonderland. In *Computers and Democracy-a Scandinavian Challenge Avebury*, ed. G. Bjerknes, P. Ehn, and M. Kyng, 279–95. Aldershot, FC.

Bjerknes, G., and T. Bratteteig. 1995. User participation and democracy: A discussion of Scandinavian research on system development. *Scand J Inf Syst* 7(1):73–98.

Bjerknes, G., P. Ehn, and M. Kyng, eds. 1987. *Computers and Democracy: A Scandinavian Challenge.* Brookfield, VT: Gower.

Björgvinsson, E., and P. A. Hillgren. 2004. On the spot experiments within healthcare. In *Proceedings of PDC 2004*, 93–101. Toronto, ON: CPSR.

Blomberg, J., J. Giacomi, A. Mosher, and P. Swenton-Wall. 1993. Ethnographic field methods and their relation to design. In *Participatory Design: Principles and Practices*, ed. D. chuler and A. Namioka. Hillsdale, NJ: Erlbaum.

Boal, A. 1974/1992. *Games for Actors and Non-Actors* (A. Jackson, Trans.). London: Routledge.

Bolton, R., eds. 1989. *The Contest of Meaning: Critical Histories of Photography*. Cambridge, MA: MIT Press.

Braa, K. 1996. Influencing qualities of information systems – future challenges for participatory design. In *Proceedings of PDC 96,* 15–24. Cambridge, MA: CPSR.

Brandt, E., and C. Grunnet. 2000. Evoking the future: Drama and props in user centered design. In *Proceedings of PDC 2000,* 11–20. New York: CPSR.

Brandt, E., and J. Messeter. 2004. Facilitating collaboration through design games. In *Proceedings of PDC 2004,* 121–30. Toronto, ON: CPSR.

Brereton, M. 2009. Designing from somewhere: A located, relational and transformative view of design. In *(Re)searching the Digital Bauhaus*, ed. T. Binder, J. Lowgren and L. Malmborg, 100–19. London: Springer.

Briefs, U., C. Ciborra, and L. Schneider. 1983. *System Design for, with, and by the Users*. Amsterdam: North-Holland.

Bruyne, E., de, and A. de Jong. 2008. *The Workplace Game: Exploring End Users' New Behaviour*. Keynote AKFEI08 conference, Las Vegas, NV.

Bunch, C. 1987. *Passionate Politics, Essays 1968–1986: Feminist Theory in Action*. New York: St. Martin's Press.

Buur, J., T. Binder, and E. Brandt. 2000. Taking video beyond "hard data" in user centred design. In *Proceedings of PDC 2000,* 1–10. New York: CPSR.

* For a more general PD bibliography, see http://www.cpsr.org/conferences/pdc98/bibliography.html.

Buzan, T., and B. Buzan. 1996. *The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential*. New York: Plume.

Bødker, S. 1990. *Through the Interface: A Human Activity Approach to User Interface Design.* Hillsdale, NJ: Erlbaum.

Bødker, S., and J. Buur. 2000. From usability lab to "design collaboratorium": Reframing usability practice. In *Proceedings of DIS'00*. New York: ACM.

Bødker, S., and J. Buur. 2002. The design collaboratorium – A place for usability design. *Trans Comput Hum Interact* 9(2):152–69.

Bødker, S., P. Ehn, M. Kyng, J. Kammersgaard, and Y. Sundblad. 1987. A UTOPIAN experience: On design of powerful computer-based tools for skilled graphic workers. In *Computers and Democracy: A Scandinavian Challenge*, ed. G. Bjerknes, P. Ehn, and M. Kyng, Brookfield, VT: Gower.

Bødker, S., and K. Grønbæk. 1991. Design in action: From prototyping by demonstration to cooperative prototyping. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Bødker, S., K. Grønbæk, and M. Kyng. 1993. Cooperative design: Techniques and experiences from the Scandinavian scene. In *Participatory Design: Principles and Practices*, ed. D. Schuler and A. Namioka. Hillsdale, NJ: Erlbaum.

Bødker, K., F. Kensing, and J. Simonsen. 2004. *Participatory IT Design: Designing for Business and Workplace Realities*. Cambridge, MA: MIT Press.

Bødker, S., J. L. Knudsen, M. Kyng, P. Ehn, and K. H. Madsen. 1988. Computer support for cooperative design. In *CSCW'88: Proceedings of the Conference on Computer Supported Cooperative Work*. Portland, OR: ACM.

Càceres, M.S. 2007, *Standardising widgets: Improving the development, distribution and deployment, accessibility, security, metadata, internationalisation, and device-independence of client-side web applications*. PhD thesis, 8/29/2007, Queensland University of Technology, Brisbane, Australia, http://datadriven.com.au/thesis/confirmation/confirmation.pdf, Verified November 28, 2011.

Cameron, M. 1998. Design for safety: Working with residents to enhance community livability. In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Carmien, S., R. DePaula, A. Gorman, and A. Kintsch. 2003. Increasing workplace independence for people with cognitive disabilities by leveraging distributed cognition among caregivers and clients. In *Proceedings of GROUP'03,* 95–104. Sanibel Island, FL: ACM.

Carrillo, R. 2000. Intersections of official script and learners' script in third space: A case study on latino families in an after-school computer program. In *Proceedings of Fourth International Conference of the Learning Sciences,* 312–13. Mahwah, NJ: Erlbaum.

Carroll, J., eds. 1995. *Scenario-Based Design for Human-Computer Interaction*. New York: Wiley.

Carroll, J. M. 2000. *Making Use: Scenario-Based Design of Human-Computer Interactions*. Cambridge, MA: MIT Press.

Carter, S., and J. Mankoff. 2005. When participants do the capturing: The role of media in diary studies. In *Proceedings of CHI 2005*. Portland, OR: ACM.

Chandler, J., A. I. Davidson, and H. Harootunian, eds. 1994. *Questions of Evidence: Proof, Practice, and Persuasion Across the Disciplines*. Chicago, IL: University of Chicago Press.

Checkland, P. 1981. *Systems Thinking, Systems Practice*. New York: Wiley.

Chin, G., K. Schuchardt, J. Myers, and D. Gracio. 2000. Participatory workflow analysis: Unveiling scientific research processes with physical scientists. In *Proceedings of PDC 2000*. New York: CPSR.

Clement, A., P. Kolm, and I. Wagner, eds. 1994. *NetWORKing: Connecting Workers in and Between Organizations*. IFIP Transactions A-38. Amsterdam: North-Holland.

Clement, A., and P. Van den Bessalaar. 1993. A retrospective look at PD projects. *Commun ACM* 36(6):29–37.

Cohene, T., R. Baecker, and E. Marziali. 2005. Designing interactive life story multimedia for a family affected by Alzheimer's disease: A case study. *CHI 2005 Extended Astracts*, 1300–03. Portland, OR. ACM.

Cook, J., and A. Light. 2006. New patterns of power and participation? Designing ICT for informal and community learning. *E-Learning* 3(1):51–61.

Cooper, A., R. Reimann, and D. Cronin. 2007. *About Face 3: The Essentials of Interaction Design*. New York, NY: Wiley.

Corburn, J. 2003. Bringing local knowledge into environmental decision making: Improving urban planning for communities at risk. *J Plann Educ Res* 22:120–33.

Costabile, M. F., D. Fogli, P. Mussio, and A. Piccinno. 2006. End-user development: The software sharing workshop approach. In *End-User Development*, ed. H. Lieberman, F. Paterno, and V. Wulf, 183–205. Dordrecht, NE: Springer.

Costabile, M. F., D. Fogli, P. Mussio, and A. Piccinno. 2007. Visual interactive systems for end-user development: A model-based design methodology. *IEEE Trans SMC Part A* 37(6):1029–46.

Cotton, L. J., A. D. Vollrath, L. Kirk, L. Froggatt, L. Mark. Lengnick-Hall, and R. Kenneth. Jennings. 1988. Employee Participation: Diverse forms and different Outcomes. *Academy of Management Review* 13(1):8–22.

Crabtree, A. 1998. Ethnography in participatory design. In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Cross, N. 2001. Designerly ways of knowing: Design discipline versus design science. *Des Issues* 17(3):49–55.

Dandavate, U., D. Steiner, and C. William. 2000. Working anywhere: Co-design through participation. In *Proceedings of CoDesigning 2000*. London: Springer.

Davies, R., S. Marcella, J. McGrenere, and B. Purves. 2004. The ethnographically informed participatory design of a PDA to support communication. In *Proceedings of ASSETS'04*. Atlanta, GA: ACM.

Davis, J. 2010. Participatory design for sustainable campus living. In *CHI 2010 Adjunct Proceedings*, 3877–82. Atlanta, GA. ACM.

de Jong, A., M. Kouprie, and E. De Bruyne. 2009. Effects of the workplace game: A case study into anticipating future behavior of office workers. In *Ergonomics and Health Aspects HCII 2009*, ed. B. T. Karsh. Berlin, Heidelberg: Springer Verlag.

Demirbilek, O., and H. Demirkan. 2004. Universal product design involving elderly users: A participatory design model. *Appl Ergon* 35:361–70.

Dewulf, G., and J. Van Meel. 2003. Democracy in design? In *Workplace Strategies and Facilities Management*, ed. R. Best, C. Langston, and G. de Valence, 281–91. Oxford, U.K.: Butterworth Heinemann.

Dewulf, G., and J. Van Meel. 2002. User participation and the role of information and communications technology. *J Corp Real Estate* 4(3):237–47.

Dingawaney, A., and C. Maier. 1994. *Between Languages and Cultures: Translation and Cross-Cultural Texts*. Pittsburgh, PA: University of Pittsburgh Press.

DiSalvo, C., A. Light, T. Hirsch, L. Goodman, and K. Hill. 2010. HCI, communities and politics. In *CHI 2010 Adjunct Proceedings*, 3151–54. Atlanta, GA. ACM.

Docherty, P., K. Fuchs-Kittowski, P. Kolm, and L. Matthiessen. 1987. *System Design for Human Development and Productivity: Participation and Beyond*. Amsterdam: North-Holland.

Dray, S. M. 1992. Understanding and supporting successful group work in software design: Lessons from IDS [Position paper]. In *Understanding and Supporting Successful Group Work in Software Design*, Workshop at CSCW '92 conference, ed. J. Karat and J. Bennett. Toronto, ON. ACM.

Druin, A. 1999. Cooperative inquiry: Developing new technologies for children with children. In *Proceedings of CHI 99*. Pittsburgh, PA: ACM.

Druin, A. 2002. The role of children in the design of new technology. *Behav Inf Technol* 21(1):1–25.

Druin, A. 2005. What children can teach us: Developing digital libraries for children. *Libr Q* 75(1):20–41.

Druin, A., H. Alborzi, A. Boltman, S. Cobb, J. Montemayor, H. Neale, M. Platner et al. 2000. Participatory design with children: Techniques, challenges, and successes. In *Proceedings of PDC 2000*. New York: CPSR.

Druin, A., B. B. Bederson, A. Rose, and A. Weeks. 2009. From New zealand to Mongolia: Co-designing and deploying a digital library for the world's children. *Chil Youth Environ Spec Issue Child Technol Environ* 19(1):34–57.

Dykstra, E. A., and R. P. Carasik. 1991. Structure and support in cooperative environments: The amsterdam conversation environment. *Int J Man Mach Stud* 34:419–34.

Ehn, P. 1988. *Work-Oriented Design of Computer Artifacts*. Falköping, Sweden: Arbetslivcentrum/Almqvist and Wiksell International.

Ehn, P. 1993. Scandinavian design: On participation and skills. In *Usability: Turning Technologies into Tools*, ed. P. S. Adler and T. A. Winograd. New York: Oxford University Press.

Ehn, P., and M. Kyng. 1987. The collective resource approach to systems design. In *Computers and Democracy: A Scandinavian Challenge*, ed. G. Bjerknes, P. Ehn, and M. Kyng. Brookfield, VT: Gower.

Ehn, P., and M. Kyng. 1991. Cardboard computers: Mocking-it-up or hands-on the future. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Ehn, P., and Å. Sanberg. 1979. *Management Control and Wage Earner Power.* Foretagsstyrning och Lontagarmakt. Falkoping: Prisma.

Ehn, P., and D. Sjögren. 1986. Typographers and carpenters as designers. In *Proceedings of Skill-Based Automation*. Karlsruhe, Germany.

Ehn, P., and D. Sjögren. 1991. From system descriptions to scripts for action. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Enquist, H., and K. Tollmar. 2008. The memory stone – a personal ICT device in health care. In *Proceedings of NordiCHI 2008*, 103–12. Lund, Sweden.

Erickson, T. 1996. Design as story-telling. *Interactions* 3(4):30–35.

Evanoff, R. 2000. The concept of "third cultures" in intercultural ethics. *Eubios J Asian Int Bioeth* 10:126–9.

Fails, J., A. Druin, and M. L. Guha. 2010. Mobile collaboration: Collaboratively reading and creating children's stories on mobile devices. In *Proceedings of Interaction Design and Children (IDC'2010).* Barcelona, Spain. ACM.

Fanderclai, T. 1995. MUDs in education: New environments, new pedagogies. *Comput Mediat Commun* 2(1):8.

Fanderclai, T. 1996. Like magic, only real. In *Wired Women: Gender and New Realities in Cyberspace*, ed. L. Cherny and E. R. Weise. Seattle, WA: Seal Press.

Floyd, C. 1987. Outline of a paradigm change in software engineering. In *Computers and Democracy: A Scandinavian Challenge*, ed. G. Bjerknes, P. Ehn, and M. Kyng. Brookfield, VT: Gower.

Floyd, C. 1993. STEPS - A methodical approach to PD. *Commun ACM* 36(6):83.

Floyd, C., W. M. Mehl, F. M. Reisin, G. Schmidt, and G. Wolf. 1989. Out of scandinavia: Alternative approaches to software design and system development. *Hum Comput Interact* 4(4):253–350.

Fowles, R. A. 2000. Symmetry in design participation in the built environment: Experiences and insights from education and practice. In *Proceedings of CoDesigning 2000*. London: Springer.

Füller, J., M. Bartl, H. Ernst, and H. Muhlbacher. 2006. Community based innovation: How to integrate members of virtual communities into new product development. *Electron Commer Res* 6(1):57–73.

Garrety, K., and R. Badham. 1998. *The Four-Dimensional Politics of Technology, or Postmodernising Participatory Design*. Presented at *Cultural Politics of Technology* workshop. Trondheim: Centre for Technology and Society.

Garzotto, F. 2008. Broadening children's involvement as design partners: From technology to "experience". In *Proceedings of the Seventh International Conference on Interaction Design and Children*, 186–93. ACM.

Gasson, S. 1995. User involvement in decision-making in information systems development. In *Proceedings of 18th IRIS*. Gjern Denmark: IRIS Association.

Gärtner, J., and I. Wagner. 1995. *Political Frameworks of Systems Design from a Cross-Cultural Perspective*. IFIP WG.9.1 Workshop. Aarhus: IFIP.

Gjersvik, R., and V. Hepsø. 1998. Using models of work practice as reflective and communicative devices: Two cases from the Norwegian offshore industry. In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Greenbaum, J. 1991. Toward participatory design: The head and the heart revisited. In *Women, Work and Computerization: Understanding and Overcoming Bias in Work and Education*, ed. I. V. Ericksson, B. A. Kitchenham, and K. J. Tijdens, 33–9. Amsterdam: Elsevier Science Publishers (North Holland).

Greenbaum, J. 1993. A design of one's own: Towards participatory design in the United States. In *Participatory Design: Principles and Practices*, ed. D. Schuler and A. Namioka. Hillsdale, NJ: Erlbaum.

Greenbaum, J. 1996. Post modern times: Participation beyond the workplace. In *Proceedings of PDC 96*. Cambridge, MA: CPSR.

Greenbaum, J., and M. Kyng. 1991. *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, NJ: Erlbaum.

Gregory, J. 2003. Scandinavian approaches to participatory design. *Int J Eng Educ* 19(1):62–74.

Grenfell, M. 1998. *Border-Crossing: Cultural Hybridity and the Rural and Small Schools Practicum*. Australian Association for Research in Education conference. Adelaide, Australia.

Gruen, D. 2000. *Storyboarding for Design: An Overview of the Process*. Cambridge, MA: Lotus Research. http://www.research.ibm.com/cambridge, under "Papers." Accessed November 28, 2011.

Gruen, D. 2001. *Stories in Design Tutorial*. IBM Make It Easy Conference. Indianapolis, IN: IBM Press.

Guha, M. L., and A. Druin. 2008. Designing with and for children of special needs: An inclusionary model. In *Interaction Design and Children (IDC'08) Workshop*. Chicago, IL. ACM.

Gumm, D. C., M. Janneck, M. Finck. 2006. Distributed participatory design – a case study. In *NordiCHI Workshop on Distributed Participatory Design*. Oslo, Norway. ACM.

Grønbæk, K. 1989. Extending the boundaries of prototyping – Toward cooperative prototyping. In *Proceedings of 12th IRIS*. Aarhus: IRIS Association.

Halskov, K., and P. Dalsgård. 2006. Inspiration card workshops. In *Proceedings of DIS 2006*, University Park, PA. ACM.

Haraway, D. 1991. Situated knowledges. In *Simians, Cyborgs, and Women: The Reinvention of Nature*, ed. D. Haraway. New York, NY: Routledge.

Harding, S. 1991. *Whose Science? Whose Knowledge? Thinking from Women's Lives*. Ithaca, NY: Cornell University Press.

Hartsock, N. 1983. The feminist standpoint. In *Discovering Reality*, ed. S. Harding and M. B. Hintikka, 283–310. Holland, Boston, London: D. Riedel Publishing Company.

Henderson, A., and M. Kyng. 1991. There's no place like home: Continuing design in use. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Heß, J., S. Offenburg, and W. Pipek. 2008. Community driven development as participation?: Involving user communities in a software design process, 31–40. In *Proc. PDC 2008*. Indiana University, Indianapolis, IN.

Hirsch, T. 2009. Communities real and imagined: Designing a communications system for zimbabwean activities. In *Proceedings of Communities and Technologies 2009*. University Park, PA. ACM.

Holmström, J. 1995. The power of knowledge and the knowledge of power: On the systems designer as a translator of rationalities. In *Proceedings of the 18th IRIS*. Göteborg: IRIS Association.

Horgan, T., M. L. Joroff, W. L. Porter, and D. A. Schön. 1998. *Excellence by Design – Transforming Workplace and Work Practice*. New York: Wiley.

Hornecker, E. 2010. Creative idea exploration within the structure of a guiding framework: The card brainstorming game. In *Proceedings of TIE 2010*, 101–8. Cambridge, MA. ACM.

Hornof, A. 2008. Working with children with severe motor impairments as design partners. In *Proceedings of the Seventh International Conference on Interaction Design and Children*, 69–72. ACM.

Howard, S., J. Carroll, J. Murphy, and J. Peck. 2002. Using 'endowed props' in scenario-based design. In *Proceedings of NORCHI 2002*. Aarhus. ACM.

Hulkko, S., T. Mattelmäki, K. Virtanen, and T. Keinonen. 2004. Mobile probes. In *Proceedings of NORCHI 2004*. Tampere: ACM.

Hultcrantz, J., and A. Ibrahim. 2002. Contextual workshops: User participation in the evaluation of future concepts. In *Proceedings of PDC 2002*. Malmö: CPSR.

Iacucci, G., C. Iacucci, and K. Kuutti. 2002. Imaging and experiencing in design – the role of performances. In *Proceedings of NORCHI 2002,* 167–176. Aarhus. ACM.

Iacucci, G., and K. Kuutti. 2002. Everyday life as a stage in creating and performing scenarios for wireless devices. *Pers Ubiquitous Comput* 6:299–306.

Irestig, M., and T. Timpka. 2002. Dynamic interactive scenario creation: A method for extending participatory design to large system development projects. In *Proceedings of PDC 2002*. Malmö: CPSR.

Isomursu, M., K. Kuutti, and S. Väinämö. 2004. Experience clip: Method for user participation and evaluation of mobile concepts. In *Proceedings of PDC 2004*. Toronto, ON: CPSR.

Iversen, O. S., and J. Buur. 2002. Design is a game: Developing design competence in a game setting. In *Proceedings of PDC 2002*. Malmö: CPSR.

Janneck, M., and D. Gumm. 2008. Bridging different use contexts in distributed participatory design. Position paper at workshop *Distributed Participatory Design*, held in conjunction with CHI 2008, Florence, IT.

Johansson, M., P. Fröst, E. Brandt, T. Binder, and J. Messeter. 2002. Partner engaged design: New challenges for workplace design. In *Proceedings of PDC 2002*. Malmö: CPSR.

Jones, C., L. McIver, L. Gibson, and P. Gregor. 2003. Experiences obtained from designing with children. In *Proceedings of Interaction Design and Children 2003: Small Users - Big Ideas*, 69–74. New York: ACM.

Jungk, R., and N. Mullert. 1987. *Future Workshops: How to Create a Desirable Future.* London: Institute of Social Invention.

Jönsson, B., and unnamed colleagues. (n.d.). *Elderly People and Design*. Lund, Sweden: Lund Institute of Technology.

Kaindl, H., L. Constantine, J. Karat, and M. J. Muller. 2001. Methods and modeling: Fiction or useful reality? (panel). In *CHI 2001 Extended Abstracts*. Seattle, WA: ACM.

Kam, M., D. Ramachandran, A. Raghavan, J. Chiu, U. Sahni, and J. Canny. 2006. Practical considerations for participatory design with rural school children in underdeveloped regions: Early reflections from the field. In *Proceedings of Interaction Design and Children 2006*, 25–32. New York: ACM.

Kankainen, T., V. Kantola, K. Mehto, and S. Tiitta. 2005. Interactive drama and user centered product concept design. In *Proceedings of Designing for User eXperience* (*DUX 2005*). San Francisco, CA, (unpaged electronic document). ACM.

Kantola, V., S. Tiitta, K. Mehto, and T. Kankainen. 2007. Using dramaturgical methods to gain more dynamic user understanding in user-centered design. In *Proceedings of Creativity and Cognition Conference*, 712–4. Washington, DC. ACM.

Kappelman, L. 1995. Measuring user involvement: A diffusion of innovation perspective. *The DATABASE Adv Inf Syst* 26(2&3):65–86.

Karasti, H., K. S. Baker, and G. C. Bowker. 2002. Ecological storytelling and collaborative scientific activities. *SIGGROUP Bull* 23(2):29–30.

Karttunen, F. 1994. *Between Worlds: Interpreters, Guides, and Survivors*. New Brunswick, NJ: Rutgers University Press.

Kensing, F., and J. Blomberg. 1998a. Participatory design: Issues and concerns. *Comput Support Coop Work* 7:167–85.

Kensing, F., and J. Blomberg, eds. 1998b. *Comput Support Coop Work* special issue on participatory design, 7(3–4):163–65.

Kensing, F., and K. H. Madsen. 1991. Generating visions: Future workshops and metaphorical design. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Kensing, F., and A. Munk-Madsen. 1993. PD: Structure in the toolbox. *Commun ACM* 36(6):78–85.

Kensing, F., J. Simonsen, and K. Bødker. 1996. MUST – A method for participatory design. In *Proceedings of PDC 96*. Cambridge, MA: CPSR.

Klein, J. T. 1996. *Crossing Boundaries: Knowledge, Disciplinarities, and Interdisciplinarities*. Charlotteville, NC: University Press of Virginia.

Klær, A., and K. H. Madsen. 1995. Participatory analysis of flexibility. *Commun ACM* 38(5):53–60.

Krabbel, A., and I. Wetzel. 1998. The customization process for organizational package information systems: A challenge for participatory design. In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Kujala, S. 2003. User involvement: A review of the benefits and challenges. *Behav Inf Technol* 22(1):1–16.

Kuutti, K., G. Iacucci, and C. Iacucci. 2002. Acting to know: Improving creativity in the design of mobile services by using performances. In *Proceedings of C&C 02*. Loughborough: ACM.

Kwok, J. Y.-C. 2004. The weight of space: Participatory design research for configuring habitable space for new arrival women in Hong Kong. In *Proceedings of PDC 2004*. Toronto, ON: CPSR.

Kyng, M. 1998. Users and computers: A contextual apporach to design of computer artifacts. *Scand J Inf Syst* 10(1,2):7–44.

Kyng, M., and L. Mathiassen. 1982. Systems development and trade union activities. In *Information Society, for Richer, for Poorer*, ed. N. Bjørn-Andersen. Amsterdam: North Holland.

Kyng, M., and L. Matthiessen, eds. 1997. *Computers in Design and Context*. Cambridge, MA: MIT Press.

Lafreniére, D. 1996. CUTA: A simple, practical, and low-cost approach to task analysis. *Interactions* 3(5):35–9.

Lanzara, G. F. 1983. The design process: Frames, metaphors, and games. In *Systems Design for, with, and by the Users*, ed. U. Briefs, C. Ciborra, and L. Schneider. Amsterdam: North-Holland.

Large, A., L. Bowler, J. Beheshti, and V. Nesset. 2007. Creating web portals with children as designers: Bonded design and the zone of proximal development. *McGill J Educ* 42(1):61–82.

Levinger, D. 1998. *Participatory Design History*. http://cpsr.org/prevsite/conferences/pdc98/history.html/view.

Light, A., and R. Luckin. 2008. *Designing for Social Justice: People, Technology, and Learning*. Futurelab. www.learningdirectorsnetwork.com/refdocs/Designing_for_Social_Justice.pdf.

Lim, Y.-K., E. Stolterman, and J. Teneberg. 2008. The anatomy of prototypes Prototypes as filters, prototypes as manifestations of design ideas. *ACM Trans Comput Hum Interact* 15(2):7:1–7:27.

Lohmann, S., S. Dietzold, P. Heim, and N. Heino. 2009. A web platform for social requirements engineering. *Software Engineering (Workshops)*, 309–15. Bonn: Köllen.

Lohmann, S., J. Ziegler, and P. Heim. 2008. Involving end users in distributed requirements engineering. In *Engineering Interactive Systems*, ed. P. Forbrig and F. Paterno, 21–8. Berlin, DE: Springer.

Louridas, P. 1999. Design as bricolage: Anthropology meets design thinking. *Des Stud* 20(6):517–35.

Luck, R. 2000. Does “inclusive design” require an inclusive design process? In *Proceedings of CoDesigning 2000*. London: Springer.

Luck, R. 2003. Dialogue in participatory design. *Des Stud* 24(6):523–35.

Lyotard, J.-F. 1984. *The Post-Modern Condition: A Report on Knowledge*. Minneapolis, MN: University of Minnesota Press.

Maarleveld, M., L. Volker, and D. J. M. van der Voordt. 2009. Measuring employee satisfaction in new offices - The WODI toolkit. *J Facil Manag* 7(3):181–97.

MacLean, A., K. Carter, L. Lovstrand, and T. Moran. 1990. User-tailorable systems: Pressing the issues with buttons. In *Proceedings of CHI ’90*. Seattle, WA: ACM.

Madsen, K. H. 1999. *Commun ACM* special issue on usability in Scandinavia and the US. 42(5).

Madsen, K. H., and P. Aiken. 1993. Experiences using cooperative interactive storyboard prototyping. *Commun ACM* 36(6):57–64.

Maher, M. L., S. J. Simoff, and G. C. Gabriel. 2000. Participatory design and communication in virtual environments. In *Proceedings of PDC 2000*. New York: CPSR.

Massimi, M. 2006. A context-aware mobile phone for remembering names and faces. Position paper for CHI 2006 workshop. In *Designing Technology for People with Cognitive Impairments*. Montréal, Canada. ACM.

Massimi, M. 2007. *Participatory Design of Mobile Phone Software for Seniors.* Master’s thesis. Toronto, Ontario, Canada: University of Toronto.

Massimi, M., and R. Baecker. 2006. Participatory design process with older users. In *Proceedings of Ubicomp 2006 Workshop on Future Networked Interactive Media Systems and Services for the New-Senior Communities*. ACM.

Massimi, M., R. Baecker, and M. Wu. 2007. Using participatory activities with seniors to critique, build, and evaluate mobile phones. In *Proceedings of ASSETS 2007*, 155–62. Tempe, AZ. ACM.

Mattelmäki, T., and K. Batarbee. 2002. Empathy probes. In *Proceedings of PDC 2002*. Malmö: CPSR.

Mazzone, E., J. Read, and R. Beale. 2008. Design with and for disaffected teenagers. In *Proceedings: NordiCHI 2008*, 290–7. New York: ACM.

McLagan, P., and C. Nel. 1995. *The Age of Participation: New Governance for the Workplace and the World*. San Francisco, CA: Berrett-Koehler.

McPhail, B., T. Costantino, D. Bruckmann, R. Barclay, and A. Clement. 1998. CAVEAT exemplar: Participatory design in a non-profit volunteer organization. *Comput Support Coop Work* 7:223–41.

Merkel, C. B., L. Xiao, U. Farooq, C. H. Ganoe, R. Lee, J. M. Carroll, and M. B. Rosson. 2004. Participatory design in community computing contexts: Tales from the field. In *Proceedings of PDC 2004*. Toronto, ON: CPSR.

Miller, D. S., M. J. Muller, and J. G. Smith. 1995. TelePICTIVE: Computer-supported collaborative GUI design for designers with diverse expertise. In *Groupware for Real Time Drawing: A Designer’s Guide*, ed. S. Greenberg, S. Haynes, and R. Rada. New York: McGraw-Hill.

Moffatt, K., J. McGrenere, B. Purves, and M. Klawe. 2004. The participatory design of a sound an image enhanced daily planner for people with aphasia. In *Proceedings of CHI 2004*. Portland, OR: ACM.

Monk, A., and S. Howard. 1998. The rich picture: A tool for reasoning about work context. *Interactions* 2:21–30.

Moore, J. M. 2003. Communicating requirements using end-user GUI constructions with argumentation. In *Proceedings of IEEE Conference on Automated Software Engineering*, 360–3. Montréal, Canada. IEEE.

Muller, M. J. 1992. Retrospective on a year of participatory design using the PICTIVE technique. In Proc. *CHI 1992,* 455–462. New York: ACM.

Muller, M. J., eds. 1994. *CPSR Newsletter* 12(3). Participatory design issue. http://cpsr.org/prevsite/publications/newsletters/old/1990s/Summer1994.txt/.

Muller, M. J. 1997a. Ethnocritical heuristics for reflecting on work with users and other interested parties. In *Computers in Context and Design*, ed. M. Kyng and L. Matthiessen. Cambridge, MA: MIT Press.

Muller, M. J. 1997b. Translation in HCI: Formal representations for work analysis and collaboration. In *Proceedings of CHI 97*. Atlanta, GA: ACM.

Muller, M. J. 1999a. *Catalogue of Scenario-Based Methods and Methodologies*. Lotus Research Technical Report 99–06. http://www.research.ibm.com/cambridge, under ”Papers.” Accessed November 28, 2011.

Muller, M. J. 1999b. *Translation in HCI: Toward a Research Agenda*. Lotus Research Technical Report 99–05. http://www.research.ibm.com/cambridge, under "Papers." Accessed November 28, 2011.

Muller, M. J. 2001. Layered participatory analysis: New development in the CARD technique. In *Proceedings of CHI 2001*. Seattle, WA: ACM.

Muller, M. J., J. L. Blomberg, K. Carter, E. A. Dykstra, J. Greenbaum, and K. Halskov Madsen. 1991. Panel: Participatory design in Britain and North America: Responses to the "Scandinavian challenge." In *Proceedings of CHI'91*. New Orleans, LA: ACM.

Muller, M. J., R. Carr, C. A. Ashworth, B. Diekmann, C. Wharton, C. Eickstaedt, and J. Clonts. 1995a. Telephone operators as knowledge workers: Consultants who meet customer needs. In *Proceedings of CHI'95*. Denver, CO: ACM.

Muller, M. J., J. D. Hallewell Haslwanter, and T. Dayton. 1997. Participatory practices in the software lifecycle. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. Landauer, and P. Prabhu. Amsterdam: Elsevier.

Muller, M. J., and S. Kuhn, eds. 1993. *Commun ACM* special issue on participatory design 36(6).

Muller, M. J., L. G. Tudor, D. M. Wildman, E. A. White, R. W. Root, T. Dayton, R. Carr, B. Diekmann, and E. A. Dykstra-Erickson. 1995b. Bifocal tools for scenarios and representations in participatory activities with users. In *Scenario-Based Design for Human-Computer Interaction*, ed. J. Carroll. New York: Wiley.

Muller, M. J., E. A. White, and D. M. Wildman. 1993. Taxonomy of PD practices: A brief practitioner's guide. *Commun ACM* 36(6):26–8.

Muller, M. J., D. M. Wildman, and E. A. White. 1994. articipatory design through games and other group exercises. In *Tutorial at CHI '94 Conference*. Boston. ACM.

Mumford, E. 1983. *Designing Human Systems for New Technology: The ETHICS Method.* Manchester, U.K.: Manchester Business School.

Mumford, E., and D. Henshall. 1979/1983. *Designing Participatively: A Participative Approach to Computer Systems Design*. Sandbach, U.K.: Manchester Business School.

Mørch, A. I., B. K. Engen, and H. R. H. Åsand. 2004. The workplace as a learning laboratory: The winding road to e-learning in a Norwegian service company. In *Proceedings of PDC 2004*. Toronto, ON: CPSR.

Naghsh, A. M., K. Danielsson, G. Fischer, T. Bratteteig, J. Blomberg, and J. A. Nocera. 2008. Distributed participatory design. In *CHI 2008 Extended Abstracts*, 3953–6. Florence, IT. ACM.

Nielsen, J., and M. Bødker. 2009. Collaborating with users: Cultural and (i)literacy challenges. In *Proceedings of OZCHI 2009*, 325–8. Melbourne, Australia. IEEE.

Nisonen, E. 1994. Women's safety audit guide: An action plan and a grass roots community development tool. *CPSR Newsletter* 12(3), Summer, 1994. http: //www.cpsr.org/publications/newsletters/issues/1994/Summer1994/nisonen.html.

Noble, A., and C. Robinson. 2000. For the love of the people: Participatory design in a community context. In *Proceedings of CoDesigning 2000*. London: Springer.

Nordichi. 2006. http://www.nordichi.org. Accessed November 28, 2011.

Noro, K., and A. S. Imada, ds. 1991. *Participatory Ergonomics*. London: Taylor & Francis.

Nygaard, K. 1975. Kunnskaps-strategi for fagbevegelsen (Knowledge strategy for trade unions). *Nordisk Forum 6* 10(2):15–27.

Nygaard, K., and P. Sørgaard. 1987. The perspective concept in informatics. In *Computers and Democracy: A Scandinavian Challenge,* ed. G. Bjerknes, P. Ehn, and M. Kyng. Brookfield, VT: Gower.

O'Connor, C., G. Fitzpatrick, M. Buchannon-Dick, and J. McKeown. 2006. Exploratory prototypes for video: Interpreting PD for a complexly disabled participant. *Proccedings of NORDICHI 2006*, 232–41. Oslo. http://www.sciweavers.org/conference/nordichi-2006.

Olsson, E. 2004. What active users and designers contribute in the design process. *Interact Comput* 16(2):377–401.

Orr, J., and N. C. Crowfoot. 992. Design by anecdote—The use of ethnography to guide the application of technology to practice. In *PDC '92: Proceedings of the Participatory Design Conference*. Cambridge, MA: CPSR.

Önder, D., and V. Der. 2007. A criteria for increasing quality in housing area: User participation. In *Proceedings of ENHR 2007*. Rotterdam.

Patton, J. W. 2000. Picturing commutes: Informant photography and urban design. In *Proceedings of PDC 2000*. New York: CPSR.

Pecknold, K. 2009. Dialogue through design: Visual communication across cultures. In *Proceedings of Creativity and Cognition*, 239–44. Berkeley, CA. ACM.

Pedell, S. 2004. Picture scenarios: An extended scenario-based method for mobile appliance design. In *Proceedings of OZCHI'04*.

Pedersen, J., and J. Buur. 2000. Games and moves: Towards innovative codesign with users. In *Proceedings of CoDesigning 2000*. London: Springer.

Peeters, M. A. G., H. F. J. M. von Tuijl, and I. M. M. Reyman. 2008. *Small Group Res* 4(39):438–67.

Pew, R. W., and A. Mavor. 2007. *Human-System Integration in the System Development Process: A New Look*. Washington, DC: National Academies Press.

Rashid, A., D. Meder, J. Wiesenberger, and A. Behm. 2006. Visual requirement specification in end-user participation. In *Proceedings of International Workshop on Multimedia Requirements Engineering*. Washington, DC. IEEE.

Reid, F. J. M., and S. E. Reed. 2000. Interaction and entrainment in collaborative design meetings. In *Proceedings of CoDesigning 2000*. London: Springer.

Rettig, M. 1994. Prototyping for tiny fingers. *Commun ACM* 37(4):21–7.

Reymen, I. M. M. J., J. M. Whyte, and C. H. Dorst. 2005. Users, designers and dilemmas of expertise. In *Proceedings of Include 2005*, London: Royal College of Art.

Robertson, T. 1996. Participatory design and participative practices in small companies. In *Proceedings of PDC 96*. Cambridge, MA: CPSR.

Robertson, T. 1998. Shoppers and tailors: Participative practices in small Australian design companies. *Comput Support Coop Work* 7(3–4):205–21.

Robertson, J. 2002. Experiences of designing with children and teachers in the StoryStation project. In *Proceedings of Interaction Design and Children 2003: Small Users - Big Ideas*, 29–41. ACM.

Saarinen, T., and M. Sääksjarvi. 1990. The missing concepts of user participation: An empirical assessment of user participation and information system success. *Scandinavian Journal of Information Systems* 2(1):25–42.

Salvador, T., and K. Howells. 1998. Focus troupe: Using drama to create common context for new product concept end-user evaluations". In *Proceedings of CHI '98.* Los Angeles: ACM.

Salvador, T., and S. Sato. 1998. Focus troupe: Mini-workshop on using drama to create common context for new product concept end-user evaluations. In *Proceedings of PDC '98*. Seattle, WA: CPSR.

Salvador, T., and S. Sato. 1999. Methods tools: Playacting and focus troupes: Theater techniques for creating quick, intense, immersive, and engaging focus group sessions." *Interactions* 6(5):35–41.

Sanders, E. B.-N. 2000. Generative tools for co-designing. In *Proceedings of CoDesigning 2000*. London: Springer.

Sanders, E. N. 2006. Scaffolds for building everyday creativity. In *Design for Effective Communications: Creating Contexts for Clarity and Meaning*, ed. J. Frascara. New York: Allworth Press.

Sanders, E. B.-N., and R. J. Branaghan. 1998. Participatory expression through image collaging: A learning-by-doing experience. In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Sanders, E. B.-N., and E. H. Nutter. 1994. Velcro-modeling and projective expression: Participatory design methods for product development. In *PDC '94: Proceedings of the Participatory Design Conference*. Chapel Hill, NC: CPSR.

Schuler, D., and A. Namioka, eds. 1993. *Participatory Design: Principles and Practices*. Hillsdale, NJ: Erlbaum.

Scrivener, S. A. R., L. J. Ball, and A. Woodcock. 2000. *Collaborative Design: Proceedings of Co-Designing 2000*. London, U.K.: Springer.

Segall, P., and L. Snelling. 1996. Achieving worker participation in technological change: The case of the flashing cursor. In *Proceedings of PDC 96*. Cambridge, MA: CPSR.

Segalowitz, M., and M. Brereton. 2009. An examination of the knowledge barriers in participatory design and the prospects for embedded research. In *Proceedings of OZCHI 2009*, 337–40. Melbourne, Australia. IEEE.

Shallwani, S., and S. Mohammed. 2007. *Community-Based Participatory Research: A Training Manual for Community-Based Researchers.* Aga Khan University Human Development Programme.

Shilton, K., N. Ramanathan, S. Reddy, V. Samanta, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. 2008. Participatory design of sensing networks: Strengths and challenges. In *Proceedings of PDC 2008*, 282–5. Bloomington, IN. Indianapolis, IN: Indiana University.

Silverstone, R., and L. Haddon. 1998. Design and the domestication of information and communication. In *Communication by Design: The Politics of Information and Communication Technologies*, ed. R. Mansell and R. Silverstone, 44–74. Oxford, U.K.: Oxford University Press.

Slater, J. 1998. Professional misinterpretation: What is participatory design? In *Proceedings of PDC 98*. Seattle, WA: CPSR.

Spencer, L. J. 1989. *Winning Through Participation: Meeting the Challenge of Corporate Chance with the Technology of Participation*. Dubuque, IA: Kendall/Hunt.

Stappers, P. J., H. van Rijn, S. C. Kirtemaker, A. E. Hennick, and F. Sleeswijk Visser. 2009. Designing for other people's strengths and motivations: Three cases using context, visions, and experiential prototypes. *Adv Eng Inf* 23(2):174–83.

Star, S. L., and J. R. Griesemer. 1989. Institutional ecology, "translations," and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907–39. *Soc Stud Sci* 19:387–420.

Suchman, L., ed. 1995. Representations of work [Special issue]. *Commun ACM* 38(9).

Suchman, L. 2002. Located accountabilities in technology production. *Scand J Inf Syst* 14(2):91–105.

Suchman, L., and R. Trigg. 1991. Understanding practice: Video as a medium for reflection and design. In *Design at Work: Cooperative Design of Computer Systems*, ed. J. Greenbaum and M. Kyng. Hillsdale, NJ: Erlbaum.

Svanæs, D., and G. Seland. 2004. Putting the users center stage: Role playing and low-fi prototyping enable end users to design mobile systems. In *Proceedings of CHI 2004*. Vienna, Austria: ACM.

Taxén, G. 2004. Introducing participatory design in museums. In *Proceedings of the Eighth Conference on Participatory Design: Artful Integration: Interweaving Media, Materials and Practices*, 204–13. New York: ACM.

Taylor, N., and K. Cheyerst. 2009. Social interaction around a rural community photo display. *Int J Hum Comput Stud* 67(12):1037–47.

Thackara, J. 2000. Edge effects: The design challenge of pervasive interface. Plenary presentation at CHI 2000.

Titlestad, O. H., K. Staring, and J. Braa. 2009. Distributed development to enable user participation: Multilevel design in the HISP network. *Scand J Inf Syst* 21(1):27–50.

Triantafyllakos, G., G. Palaigeorgiou, and I. A. Tsoukalas. 2010. Fictional characters in participatory design sessions: Introducing the "design alter egos" technique. *Interact Comput* 22(3):165–75.

Trigg, R. H. 2000. From sandbox to "fundbox": Weaving participatory design into the fabric of a busy non-profit. In *Proceedings of PDC 2000*. New York: CPSR.

Tscheligi, M., S. Houde, A. Marcus, K. Mullet, M. J. Muller, and R. Kolli. 1995. Creative prototyping tools: What interaction designers really need to produce advanced user interface concepts. In *CHI'95 Conference Companion*. Denver, CO: ACM.

Tschudy, M. W., E. A. Dykstra-Erickson, and M. S. Holloway. 1996. PictureCARD: A storytelling tool for task analysis. In *PDC'96 Proceedings of the Participatory Design Conference*. Cambridge, MA: CPSR.

Tudor, L. G., M. J. Muller, T. Dayton, and R. W. Root. 1993. A participatory design technique for high-level task analysis, critique, and redesign: The CARD method. In *Proceedings of the HFES'93*. Seattle, WA: Human Factors and Ergonomics Society.

Törpel, B., and M. Poschen. 2002. Improving infrastructures by transforming narratives. In *Proceedings of PDC 2002*. Malmö: CPSR.

Vaajakallio, K., and T. Mattelmäki. 2007. Collaborative design exploration: Envisioning future practices with make tools. In *Proceedings of Designing Pleasurable Products and Interfaces*, 223–38. Helsinki, Finland. ACM.

Van den Besselaar, P., A. Clement, and P. Jaervinen. 1991. *Information System, Work and Organization Design*. Amsterdam: North-Holland.

van den Besselaar, P., J. Greenbaum, and P. Mambrey. 1996. Unemployment by design: Participatory design and the changing structure of the workforce in the information society. In *Proceedings of PDC 96*. Cambridge, MA: CPSR.

von Hippel, E. 2002. Horizontal *Innovation Networks - By and For Users*. MIT Sloan School of Management Working Paper No 4366-02 (web.mit.edu/evhippel/www/papers/UserInnovNetworksMgtSci.pdf).

Wakkary, R., and K. Tanenbaum. 2009. A sustainable identity: The creativity of an everyday designer. In *Proceedings of CHI 2009*, 365–74. Boston, MA. New York: ACM.

Wang, C., M. A. Burris, and X. Y. Ping. 1996. Chinese village women as visual anthropologists: A participatory approach to reaching policymakers. *Soc Sci Med* 42(10):1391–440.

Warr, A. 2006. Situated and distributed design. In *NordiCHI Workshop on Distributed Participatory Design*. Oslo, Norway. ACM.

Warr, A., and E. O'Neill. 2005. Understanding design as a social creative process. In *Proceedings of Conference on Creativity and Cognition*, 12–15. London, United Kingdom. ACM.

Winner, L. 1994. Political artifacts in Scandinavia: An american perspective. *Scand J Inf Syst* 6(2):85–94.

Winters, N., and Y. Mor. 2008. IDR: A participatory methodology for interdisciplinary design in technology enhanced learning. *Comput Educ* 50(2):579–600.

Wixon, D., and J. Ramey, eds. 1996. *Field Methods Casebook for Software Design*. New York: Wiley.

Wu, M., R. Baecker, and B. Richards. 2005. Participatory design of an orientation aid for amnesics. In *Proceedings of CHI 2005*. Portland, OR: ACM.

Wu, M., B. Richard, and R. Baecker. 2004. Participatory design with individuals who have amnesia. In *Proceedings of PDC 2004*. Toronto, ON: CPSR.

Xie, B., A. Druin, J. Fails, S. Massey, E. Golub, S. Franckel, and K. Schneider. (In press). Connecting generations: Developing co-design methods for older adults and children. *Behav Inf* Technol.

Yamauchi, Y. 2009. Power of peripheral designers: How users learn to design. In *Proceedings of 4th International Conference on Design Science Research in Information Systems and Technology*. Philadelphia, PA, (unpaged electronic edition). ACM.

Yu, Y., and Z. Liu. 2006. Integrated scenario-based design method for inclusive online teaching system. In *Proceedings of International Conference on Computational Intelligence for Modeling Control and Automation*. Sydney, Australia. IEEE.

50 Unified User Interface Development
A Software Refactoring Perspective

Anthony Savidis and Constantine Stephanidis

CONTENTS

50.1 INTRODUCTION

This chapter discusses the unified user interface development paradigm as an effective software engineering recipe for user interfaces that can be automatically adapted to individual end users and contexts of use. The distinctive procedural, representational, and programming properties of this development discipline are highlighted, qualifying it as a plausible and cost-effective approach toward the goal of developing automatic user interface personalization. The method conveys a fundamentally new perspective into the development of user interfaces, emphasizing a principled, systematic and evolutionary approach toward coping with diversity, by progressively encapsulating and coordinating in an extensible development structure all alternative interaction artifacts.

Finally, we show that runtime adaptation (adaptive behavior) is essentially a cross-cutting system concern for interactive software applications that can be gradually accommodated a posteriori through a software refactoring process. The latter represents a significant methodological leap, since it reveals that adaptive behavior may be introduced within originally nonadaptive systems through a sequence of systematic, architecture-preserving, source code transformations.

50.1.1 Automatic User Interface Adaptation

The notion of automatic user interface adaptation reflects the capability of interactive software to adapt during runtime to the individual end user and to the particular context of use by delivering the most appropriate interactive experience. The storage location, origin, and format of user-oriented information may vary. For example, information may be stored in profiles, indexed by unique user identifiers, extracted from user-owned cards, entered by the user in an initial interaction session, or inferred by the system through continuous interaction monitoring and analysis. Additionally, usage-context information, such as user location, environment noise, network bandwidth, and so forth, is normally provided by special purpose equipment, like sensors, or system-level software. To support optimal interface delivery for individual user- and usage-context attributes, it is required that for any given user task or group of user activities, the implementations of the alternative best fit interface components are either implementationally encapsulated or appropriately locatable (e.g., remote components, downloadable plug-ins, dynamically linked libraries, etc.).

50.1.2 The Concept of Unified User Interfaces

A unified user interface is the interaction-specific software of software applications or services, which is capable of self-adapting to the characteristics of the individual end user and context of use. Such an adaptation may reflect varying patterns of interactive behavior, at the physical, syntactic, or semantic level of interaction, to accommodate specific user- and context-oriented parameters. Practically speaking, from the end-user point of view, a unified user interface is actually an interface that can automatically adapt to the individual user attributes (e.g., requirements, abilities, and preferences), as well as to the particular characteristics of the usage context (e.g., computing platform, peripheral devices, interaction technology, and surrounding environment). Therefore, a unified user interface realizes the combination of the following:

- User-adapted behavior, such as the automatic delivery of the most appropriate user interface for the particular end user (requires user awareness).
- Usage-context-adapted behavior, such as the automatic delivery of the most appropriate user interface for the particular situation of use (requires usage context awareness).
- Hence, the characterization unified does not have any particular behavioral connotation, at least as seen from an end-user perspective. Instead, the notion of unification reflects the specific software engineering strategy needed to accomplish this behavior, emphasizing the proposed systematic development-oriented discipline. More specifically, to realize this form of adapted behavior, a unified user interface reflects the following fundamental development properties.
- It encapsulates alternative dialogue patterns (e.g., implemented dialogue artifacts), for various dialogue design contexts (e.g., a subtask, a primitive user action, a visualization), appropriately associated with the different values of user- and usage-context-related attributes. The need for such alternative dialogue patterns may only be dictated by the user interface design process, when, given any particular design context, for different user- and usage-context attribute values, alternative design artifacts are needed to accomplish optimal interaction.
- It implementationally encapsulates representation schemes for user- and usage-context parameters, internally utilizing user- and usage-context information resources (e.g., repositories, servers), to extract or to update user- and usage-context information.
- It encapsulates the necessary user interface design knowledge and decision-making capability for activating, during runtime, the most appropriate dialogue patterns (e.g., interactive software components), according to particular instances of user- and usage-context attributes.

The distinctive property of unified user interfaces to encapsulate alternative, mutually exclusive, user interface design artifacts, each purposefully designed and implemented as an optimal alternative for its corresponding target attributes of the user- and usage-context models, assuming a particular design context, constitutes one of the main contributions of this research work within the user interface software engineering field.

50.2 UNIFIED USER INTERFACE DESIGN

50.2.1 The Design Problem

Universal design in human–computer interaction (HCI) reflects the principle to address potentially all users and usage contexts—anyone, anyplace, anytime. Its main objective is to ensure that each end user is given the most appropriate interactive experience by supporting both accessible and high-quality interaction. In this context, to accommodate genuinely universal design in the production process of software application and services, two key issues need to be optimally addressed. The first is scientific, primarily concerning the way the particular problem is to be technically resolved, while the second is cost-specific, reflecting the criticality for cost-effective and economically viable solutions. Clearly, producing and enumerating distinct user interface designs through the conduct of multiple design processes is an impractical solution, since the overall cost for managing in parallel such a large number of independent design processes, as well as for transforming each produced user interface version into a target software implementation, would be unacceptable both for the design and the software implementation phases. Instead, a design process is required which may lead to a single design outcome that appropriately links and organizes the differentiating aspects of the resulting interactive application, around common abstract design structures and patterns, making it far easier to (a) map to a target software system implementation and (b) maintain, update, and extend the design itself.

The need for introducing alternative design artifacts, even for the same specific design context (such as a particular subtask), emerges from the fact that, in universal design, the design problem encompasses largely varying parameters, such as user- and usage-context parameters. Consequently, when designing for any particular dialogue design context, it is likely that differentiating values of those problem parameters dictate the design of diverse dialogue artifacts. This issue introduces two important requirements for pursuing a suitable design method.

The first is that such a method should offer the capability to associate multiple alternative dialogue artifacts to a particular single design context, due to the varying design problem parameters, by enabling the unambiguous association of each alternative artifact with its corresponding values of the problem parameters.

The second is that the method should emphasize capturing of the more abstract structures and patterns inherent within user interface designs, enabling the hierarchical incremental specialization toward the lower physical level of interaction, making it possible to introduce alternative dialogue patterns as close to the physical design as possible. This will make it far easier for the design space to be updated and to evolve, since modifications and extensions due to the consideration of additional values of the problem parameters (e.g., considering new user- and usage-context attribute values) can be applied locally closer to the lower levels of the design, without affecting the rest of the design space.

To demonstrate briefly the need for supporting alternative dialogue artifacts for the same design context, an example from a real-life application will be used. The AVANTI web browser (Stephanidis et al. 2001) was developed to enable web access by supporting adaptation to the particular user as well as to the context of use. During the user interface design phase, while concentrating on the design context of the link dialogue task, alternative designs have been dictated, due to varying user- and usage-context parameters considered, as shown in Figure 50.1.

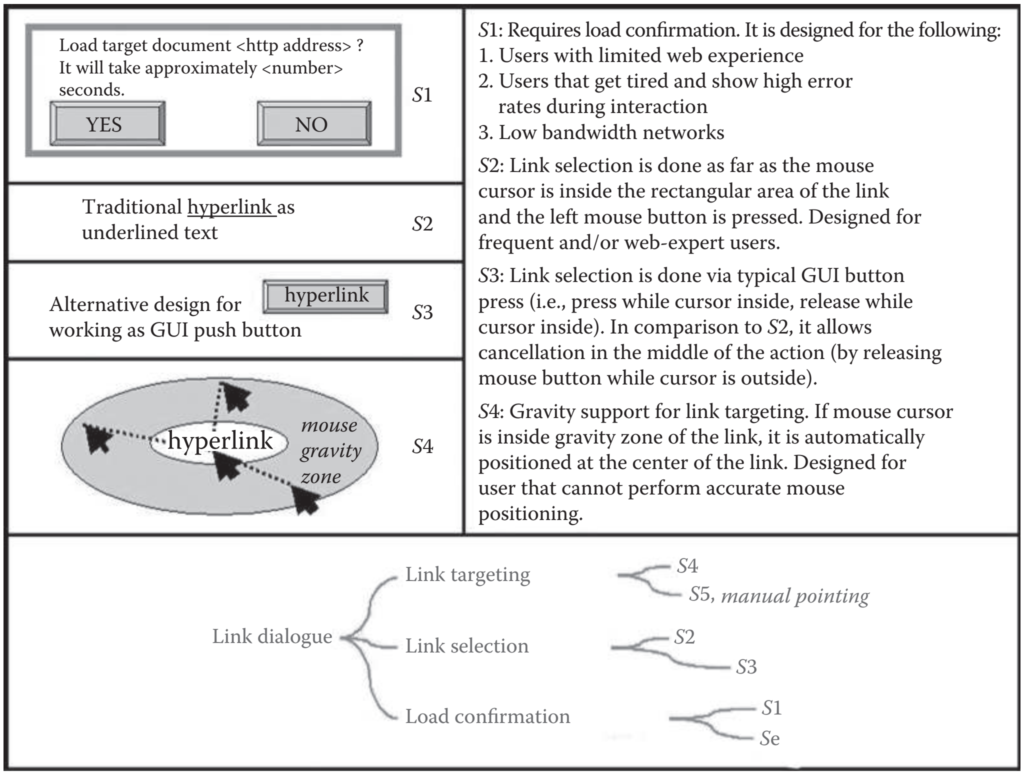
Since the differing artifacts have been part of the final AVANTI web browser user interface design, the design representation formalism is needed to enable their copresence within the resulting design space by clearly associating each artifact to the link-selection task and its corresponding values of the user- and usage-context parameters. A loose design notation is used in Figure 50.1 to show hierarchical task analysis (subtask sequencing is omitted for clarity), as well as the need for alternative incarnations of a single task (e.g., styles S2/S3 for link selection, styles S1/Se for load confirmation, and styles S4/S5 for link targeting). Following this example taken from the AVANTI web browser, during runtime, depending on the particular end user and usage-context attribute values, the appropriate corresponding implemented artifacts will only have to be activated.

50.2.2 Polymorphic Task Hierarchies

A polymorphic task hierarchy combines three fundamental properties: (a) hierarchical decomposition, (b) polymorphism, and (c) task operators. The hierarchical decomposition adopts the original properties of hierarchical task analysis for incremental decomposition of user tasks to lower-level actions. The polymorphism property provides the design differentiation capability at any level of the task hierarchy, according to particular user- and usage-context attribute values. Finally, task operators, which are based on the communicating sequential processes (CSP) language for describing the behavior of reactive systems (Hoare 1978), enable the expression of dialogue control flow formulae for task accomplishment. Those specific operators, taken from the domain of reactive systems and process synchronization, have been selected due to their appropriateness in expressing temporal relationships of user actions and tasks. However, designers may freely use additional operators as needed (e.g., the set is not closed) or may choose to document dialogue sequencing and control outside the task-structure in natural language, when it engages more comprehensive algorithmic logic (e.g., consider the verbally documented precondition if the logged user is a guest, no sign-in is required, else the access privileges should be checked and the sign-in dialogue is activated before chat).

The concept of polymorphic task hierarchies is illustrated in Figure 50.2. Alternative task decomposition is called a “decomposition style,” or simply a “style,” and is to be given by designers an appropriate descriptive name. Alternative task sub-hierarchies are attached to their respective styles. The example polymorphic task hierarchy of Figure 50.2 indicates the way two alternative dialogue styles for a “Delete File” task can be designed—one exhibiting direct manipulation properties with object-function syntax (e.g., the file object is selected prior to operation to be applied) with no confirmation, the other realizing modal dialogue with a function-object syntax (e.g., the delete function is selected, followed by the identification of the target file) and confirmation.

Additionally, the example demonstrates the case of physical specialization. Since selection is an abstract task, it is possible to design alternative ways for physically instantiating the selection dialogue (see Figure 50.2, lower-part), via scanning techniques for motor-impaired users, via three-dimensional (3D) hand pointing on 3D auditory cues for blind people, via enclosing areas (e.g., irregular rubber banding) for sighted users, and via Braille output and keyboard input for deaf and blind users. The unified user interface design



*S*1: Requires load confirmation. It is designed for the following:
1. Users with limited web experience
2. Users that get tired and show high error rates during interaction
3. Low bandwidth networks

*S*2: Link selection is done as far as the mouse cursor is inside the rectangular area of the link and the left mouse button is pressed. Designed for frequent and/or web-expert users.

*S*3: Link selection is done via typical GUI button press (i.e., press while cursor inside, release while cursor inside). In comparison to *S*2, it allows cancellation in the middle of the action (by releasing mouse button while cursor is outside).

*S*4: Gravity support for link targeting. If mouse cursor is inside gravity zone of the link, it is automatically positioned at the center of the link. Designed for user that cannot perform accurate mouse positioning.

FIGURE 50.1 Designing alternative artifacts for the Link dialogue task. *S*e is used to indicate “empty” (i.e., no load confirmation dialogue supported). *S*5 is the typical manual link targeting GUI dialogue through the mouse.

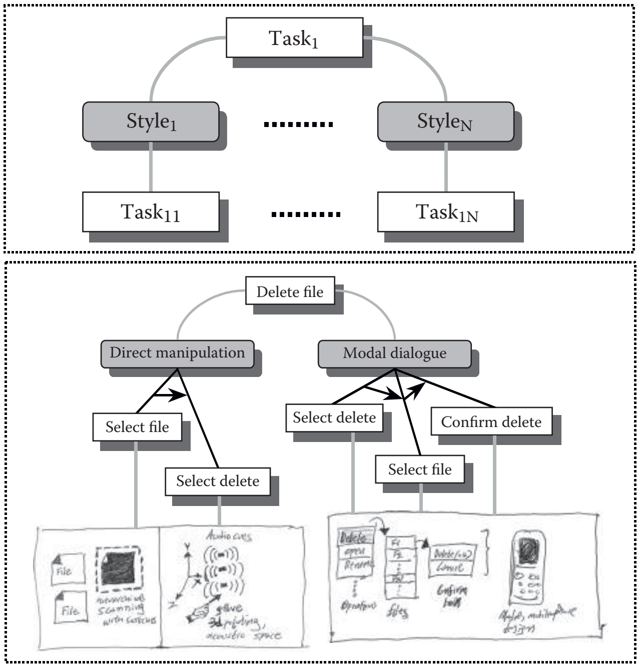


FIGURE 50.2 The polymorphic task hierarchy concept, where alternative decomposition “styles” are supported (upper part), and an exemplary polymorphic decomposition, which includes physical design annotation (lower part).

method does not require the designer to follow the polymorphic task decomposition all the way down the user-task hierarchy, until primitive actions are met. A nonpolymorphic task can be specialized at any level, following any design method chosen by the interface designer. For instance, in Figure 50.2 (lower part), graphical mock-ups are used to describe each of the alternative physical instantiations of the abstract selection task. It should be noted that the interface designer is not constrained to using a particular model, such as CSP operators, for describing user actions for device-level interaction (e.g., drawing, drag-and-drop, concurrent input). Instead, an alternative may be preferred, such as an event-based representation, such as ERL (Hill 1986) or UAN (Hartson et al. 1990).

As discussed in more detail in the subsequent sections, design polymorphism entails a decision-making capability for context-sensitive selection among alternative artifacts to assemble a suitable interface instance, while task operators support temporal relationships and access restrictions applied to the interactive facilities of a particular interface instance.

50.2.3 The Conduct of the Method

In the unified user interface design method, there are three categories of design artifacts, all of which are subject to polymorphism on the basis of varying user- and usage-context parameter values. These three categories are as follows (see Figure 50.3):

1. User tasks, relating to what the user has to do; user tasks are the center of the polymorphic task decomposition process.
2. System tasks, representing what the system has to do, or how it responds to particular user actions (e.g., feedback); in the polymorphic task decomposition process, system tasks are treated in the same manner as user tasks.
3. Physical design that concerns the various physical interface components on which user actions corresponding to the associated user task are to be performed; the physical structure may also be subject to polymorphism.

System tasks and user tasks may be freely combined within task formulas, defining how sequences of user-initiated actions and system-driven actions interrelate. The physical design, providing the interaction context, is always associated with a particular user or system task. It provides the physical dialogue pattern associated to a task-structure definition. Hence, it plays the role of annotating the task hierarchy with physical design information. An example of such annotation is shown in Figure 50.2, where the physical designs for the "Select delete" task are explicitly depicted.

In some cases, given a particular user task, there is a need for differentiated physical interaction contexts, depending on user- and usage-context parameter values. Hence, even though the task decomposition is not affected (e.g., the same user actions are to be performed), the physical design may have to be altered. One such representative example is relevant to changing particular graphical attributes on the basis of ethnographic user attributes. For instance, Marcus (1996) discussed the choice of different iconic representations, background patterns, visual message structure, and so forth on the basis of cultural background. User tasks, and in certain cases, system tasks, need not always be related to physical interaction, but may represent abstraction on either user or system actions. For instance, if the user has to perform a selection task, then, clearly, the physical means of performing such a task are not explicitly defined, unless the dialogue steps to perform selection are further decomposed.

This notion of continuous refinement and hierarchical analysis, starting from higher level abstract artifacts, and incrementally specializing towards the physical level of interaction is fundamental in the context of hierarchical behavior analysis, either regarding tasks that humans have to perform (Johnson et al. 1988) or when it concerns functional system design (Saldarini 1989). At the core of the unified user interface design method is the polymorphic task decomposition process, which follows the methodology of abstract task definition and incremental specialization, where tasks may be hierarchically analyzed through various alternative schemes. In such a recursive process, involving tasks ranging from the abstract task level to specific physical actions, decomposition is applied either in a traditional unimorphic fashion or by means of alternative styles. The overall process is illustrated in Figure 50.4; the decomposition starts from abstract or physical task design, depending on whether top-level user

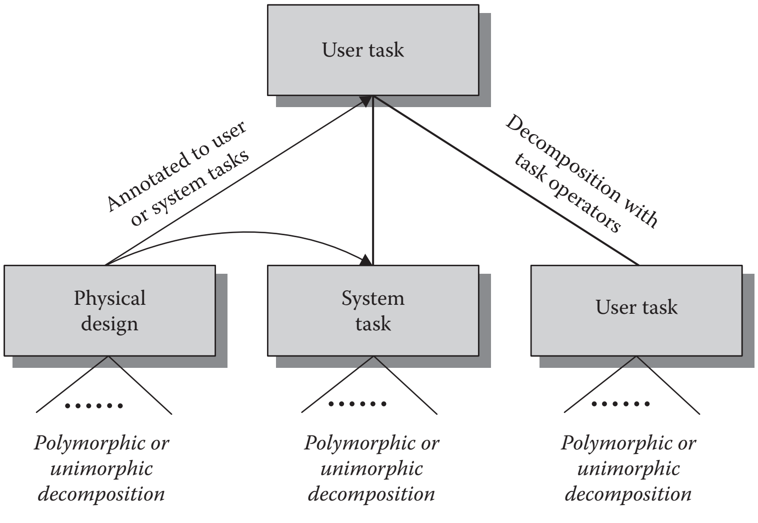

User task
Annotated to user or system tasks
Decomposition with task operators
Physical design
System task
User task
Polymorphic or unimorphic decomposition
Polymorphic or unimorphic decomposition
Polymorphic or unimorphic decomposition

FIGURE 50.3 The three artifact categories in the unified user interface design method, for which polymorphism may be applied, and how they relate to each other.

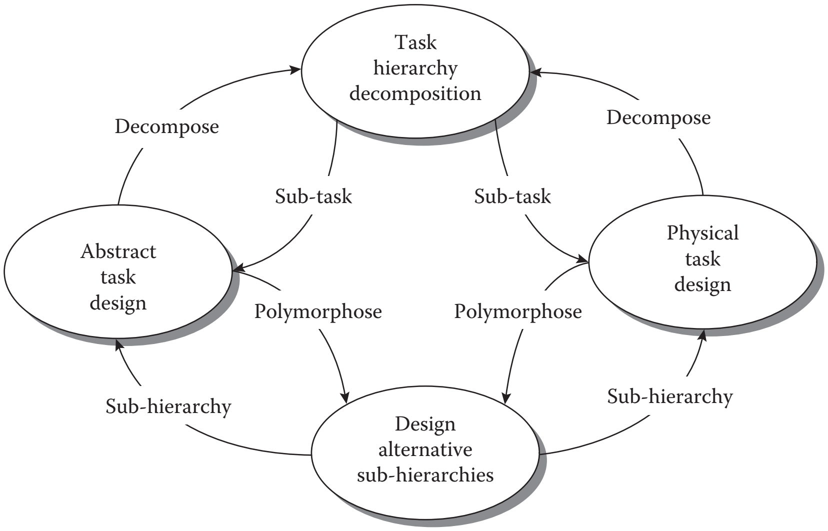


FIGURE 50.4 The polymorphic task decomposition process in the unified user interface design method.

tasks can be defined as being abstract or not. Next follows the description of the various transitions (e.g., design specialization steps) from each of the four states illustrated in the process state diagram of Figure 50.4.

50.2.3.1 Transitions from the Abstract Task Design State

An abstract task can be decomposed either in a polymorphic fashion, if user- and usage-context attribute values pose the necessity for alternative dialogue patterns, or in a traditional manner, following a unimorphic decomposition scheme. In the case of a unimorphic decomposition scheme, the transition is realized via a decomposition action, leading to the task hierarchy decomposition state. In the case of a polymorphic decomposition, the transition is realized via a polymorphose action, leading to the alternative sub-hierarchies design state.

50.2.3.2 Transitions from the Alternative Sub-Hierarchies Design State

Reaching this state means that the required alternative dialogue styles have been identified, each initiating a distinct sub-hierarchy decomposition process. Hence, each such sub-hierarchy initiates its own instance of polymorphic task decomposition process. While initiating each distinct process, the designer may start either from the abstract task design state or from the physical task design state. The former is pursued if the top-level task of the particular sub-hierarchy is an abstract one. In contrast, the latter option is relevant in case the top-level task explicitly engages physical interaction issues.

50.2.3.3 Transitions from the Task Hierarchy Decomposition State

From this state, the subtasks identified need to be further decomposed. For each subtask at the abstract level, there is a subtask transition to the abstract task design state. Otherwise, if the subtask explicitly engages physical interaction means, a subtask transition is taken to the physical task design state.

50.2.3.4 Transitions from the Physical Task Design State

Physical tasks may be further decomposed either in a unimorphic fashion or in a polymorphic fashion. These two alternative design possibilities are indicated by decompose and polymorphose transitions, respectively.

50.2.4 Relationships among Alternative Styles

The need for alternative styles emerges during the design process when it is identified that there are particular user- and usage-context attribute values not addressed by the dialogue artifacts that have already been designed. Starting from this observation, one could argue that all alternative styles, for a particular polymorphic artifact, are mutually exclusive to each other (in this context, exclusion means that, at runtime, only one of those styles may be active).

However, cases in which it is meaningful to make artifacts belonging to alternative styles exist, concurrently available in a single adapted interface instance. A typical case concerns file management tasks, where two alternative but compatible artifacts may coexist during runtime, such as a direct-manipulation one and a command-based one, such as task-level multimodality. In the unified user interface design method, four design relationships between alternative styles are distinguished (see Figure 50.5), defining whether alternative styles may be concurrently present at runtime. These four fundamental relationships reflect pragmatic, real-world design scenarios.

50.2.4.1 Exclusion

The exclusion relationship is applied when the various alternative styles are deemed usable only within the space of their target user- and usage-context attribute values. For instance, assume that two alternative artifacts for a particular subtask are being designed, aiming to address the user-expertise attributes: one targeted to users qualified as novice and the other targeted to expert users. Then, these two are defined

Exclusion *Relates many styles.* Only one from the alternative styles may be present.

Compatibility *Relates many styles.* Any of the alternative styles may be present.

Substitution *Relates two groups of styles together.* When the second is made "active" at runtime, the first should be "deactivated."

Augmentation *Relates one style with a group of styles.* On the presence of any style from the group at runtime, the single style may be also "activated."

FIGURE 50.5 Design relationships among alternative styles and their runtime interpretation.

to be mutually exclusive to each other, since it is probably meaningless to activate concurrently both dialogue patterns. For example, at runtime a novice user might be offered a functionally simple alternative of a task, where an expert user would be provided with additional functionality and greater freedom in selecting different ways to accomplish the same task.

50.2.4.2 Compatibility

Compatibility is useful among alternative styles for which the concurrent presence during interaction allows the user to perform certain actions in alternative ways, without introducing usability problems. The most important application of compatibility is for task multimodality, as it has been previously discussed for the file management tasks.

50.2.4.3 Substitution

Substitution has a very strong connection with adaptivity techniques. It is applied in cases where, during interaction, it is decided that some dialogue patterns need to be substituted by others. For instance, the ordering and the arrangement of certain operations may change on the basis of monitoring data collected during interaction, through which information such as frequency of use and repeated usage patterns can be extracted. Hence, particular physical design styles would need to be cancelled, while appropriate alternatives would need to be activated. This sequence of actions, for example, cancellation followed by activation, is the realization of substitution. Thus, in the general case, substitution involves two groups of styles: some styles are cancelled and substituted by other styles that are activated afterwards.

50.2.4.4 Augmentation

Augmentation aims to enhance the interaction with a particular style that is found to be valid but not sufficient to facilitate the user's task. To illustrate this point, assume that, during interaction, the user interface detects that the user is unable to perform a certain task. This would trigger an adaptation (in the form of adaptive action) aiming to provide task-sensitive guidance to the user. Such an action should not aim to invalidate the active style (by means of style substitution), but rather to augment the user's capability to accomplish the task more effectively by providing informative feedback. Such feedback can be realized through a separate but compatible style. Therefore, it follows that the augmentation relationship can be assigned to two styles when one can be used to enhance the interaction while the other is active. Thus, for instance, the adaptive prompting dialogue pattern, which provides task-oriented help, may be related via an augmentation relationship with all alternative styles (of a specific task), provided that it is compatible with them.

50.3 UNIFIED INTERFACE ENGINEERING

In the context of a unified user interface, upon startup and during runtime, the software user interface relies on the particular user and context profiles to assemble the eventual interface on the fly, collecting and gluing together the constituent interface components required for the particular end user and usage context. Such constituent components are the alternative artifacts identified during the user interface design process, which need to be appropriately transformed in the development phase to an implementation form.

Effectively, a unified user interface consists of runtime components, each with a distinctive role in performing at runtime a type of an interface assembly process, by selecting the most appropriate dialogue patterns from the available implemented design space (e.g., the organized collection of all dialogue artifacts produced during the design phase). Examples of such intelligent selection and assembly of user interface components for adapted interaction delivery are provided in Figures 50.6 and 50.7, taken from the AVANTI Project (Stephanidis et al. 2001).

A unified user interface does not constitute a monolithic software system but becomes a distributed architecture consisting of independent intercommunicating components, possibly implemented with different software methods/tools and residing at different physical locations. These components co-operate to perform adaptation according to the individual end-user attributes and the particular usage context. At runtime, the overall adapted interface behavior is realized by two complementary classes of system-initiated actions:

1. Adaptations driven from initial user and context information, acquired without performing interaction monitoring analysis (e.g., what is known before starting observing the user or the usage context)
2. Adaptations decided on the basis of information inferred or extracted by performing interaction-monitoring analysis (e.g., what is learned by observing the user or the usage context)

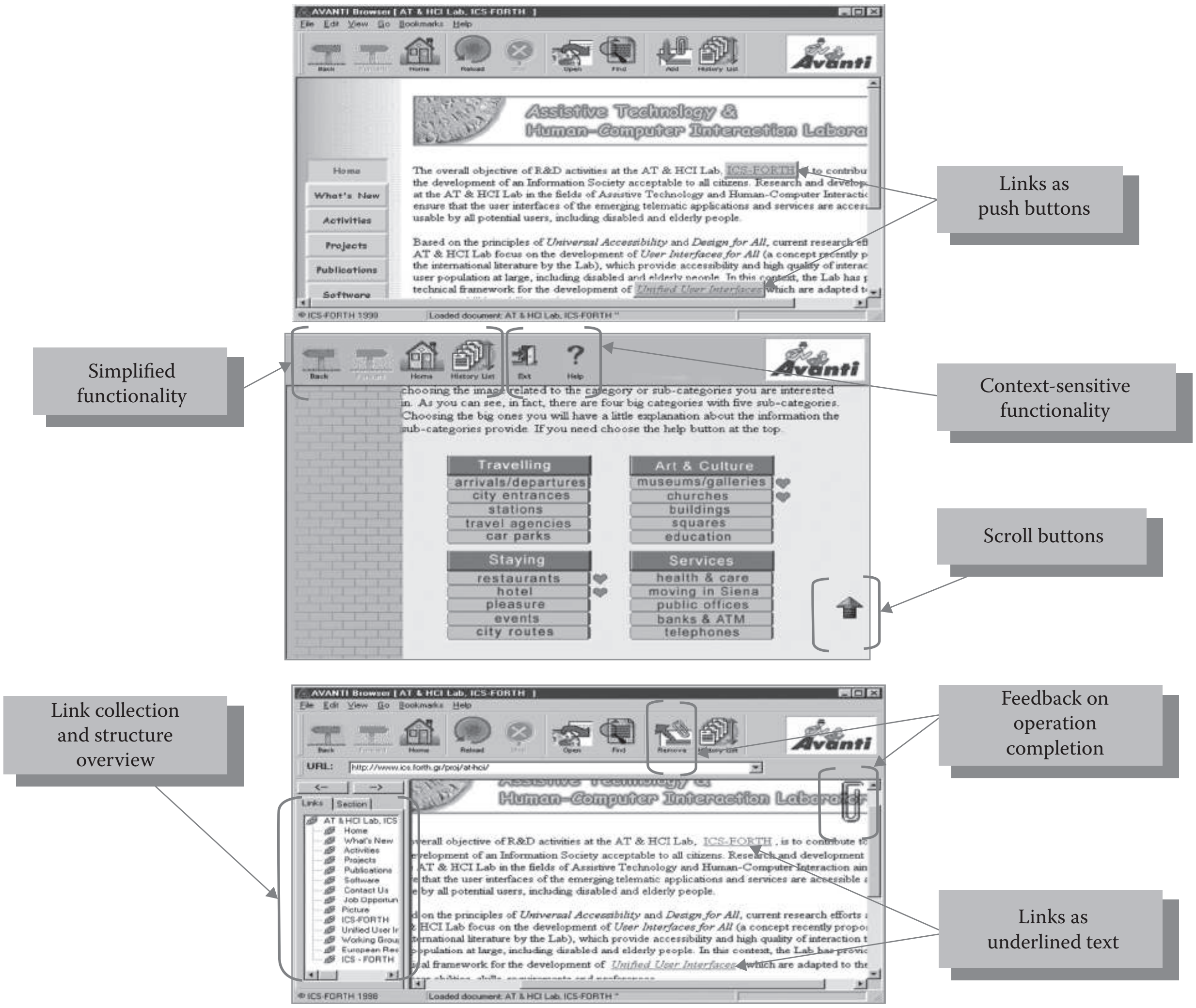


FIGURE 50.6 Three different interface versions of the AVANTI browser produced by adaptation-oriented selection of different alternative styles.

The former behavior is referred to as "adaptability" (e.g., initial automatic adaptation, performed before initiation of interaction), reflecting the capability of the interface to proactively and automatically tailor itself to the attributes of each individual end user. The latter behavior is referred to as "adaptivity" (e.g., continuous automatic adaptation) and characterizes the capability of the interface to cope with the dynamically changing/evolving characteristics of users and usage contexts. Adaptability is crucial to ensure accessibility, since it is essential to provide, before initiation of interaction, a fully accessible interface instance to each individual end user. Adaptivity can be applied only on accessible running interface instances (e.g., ones with which the user is capable of performing interaction), since interaction monitoring is required for the identification of changing or emerging decision parameters that may drive dynamic interface enhancements. The complementary roles of adaptability and adaptivity are depicted in Figure 50.8. This fundamental distinction is made due to the different runtime control requirements between those two key classes of adaptation behaviors, requiring different software engineering policies.

50.3.1 Architectural Overview

In this section, the runtime architecture for unified user interfaces will be discussed, providing an outline of the adopted architectural components with information regarding (a) the functional role, (b) the runtime behavior, (c) the encapsulated context, and (d) the implementation method. The components of the unified user interface architecture are the following (see Figure 50.9):

- User-information server (UIS)
- Context parameters server
- Decision-making component (DMC)
- Dialogue patterns component (DPC)

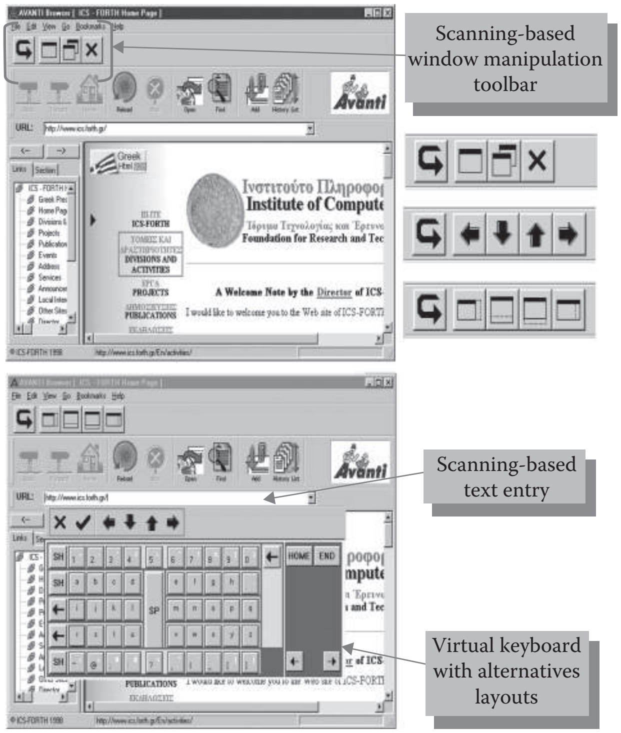

Scanning-based window manipulation toolbar
Scanning-based text entry
Virtual keyboard with alternatives layouts

FIGURE 50.7 Alternative augmentation-oriented styles for motor-impaired user access activated at runtime in the AVANTI browser.

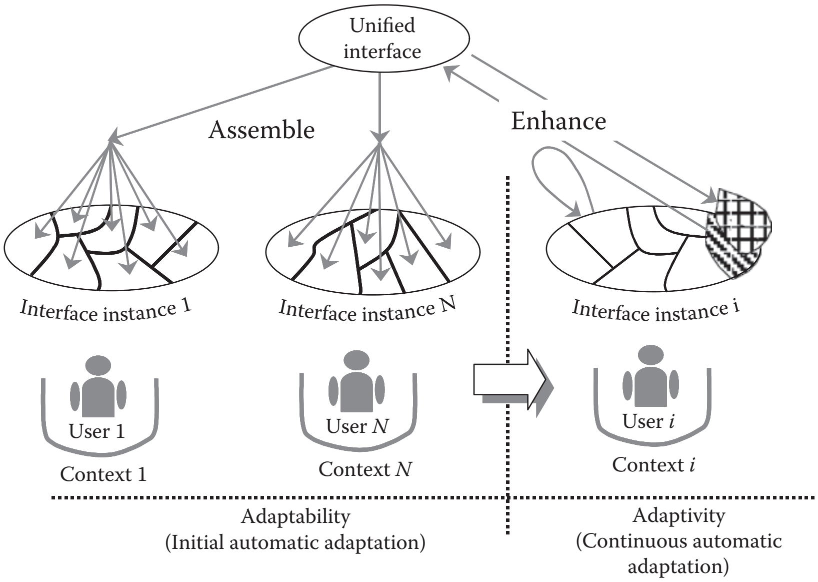


FIGURE 50.8 The complementary roles of adaptability (left) and adaptivity (right) as realized in unified user interfaces to provide user- and usage-context-adapted-behavior.

50.3.1.1 User-Information Server

Its functional role is to supply user attribute values (1) known offline, without performing interaction-monitoring analysis (e.g., motor/sensory abilities, age, nationality, etc.), and (2) detected online, from real-time interaction-monitoring analysis (e.g., fatigue, loss of orientation, inability to perform the task, interaction preferences, etc.). During runtime, it plays a twofold role: (1) it constitutes a server that maintains and provides information regarding individual user profiles and (2) it encompasses user-representation schemes, knowledge processing components, and design information dynamically to detect user properties or characteristics. This component may need to use alternative ways of representing user-oriented information. In this sense, a repository of user profiles serves as a central database of individual user information (e.g., registry). In many cases, a profile structure as a typical list of typed attributes will suffice shown; this model, though quite simple, is proved in real practice to be very powerful and flexible (can be stored in a database, thus turning the profile manager to a remotely accessed database).

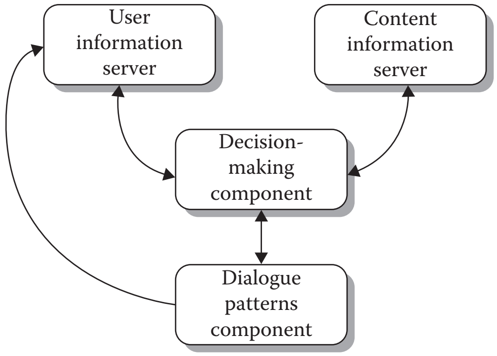

User information server
Content information server
Decision-making component
Dialogue patterns component

FIGURE 50.9 The four basic components of the unified user interface macroarchitecture outlining runtime communication links.

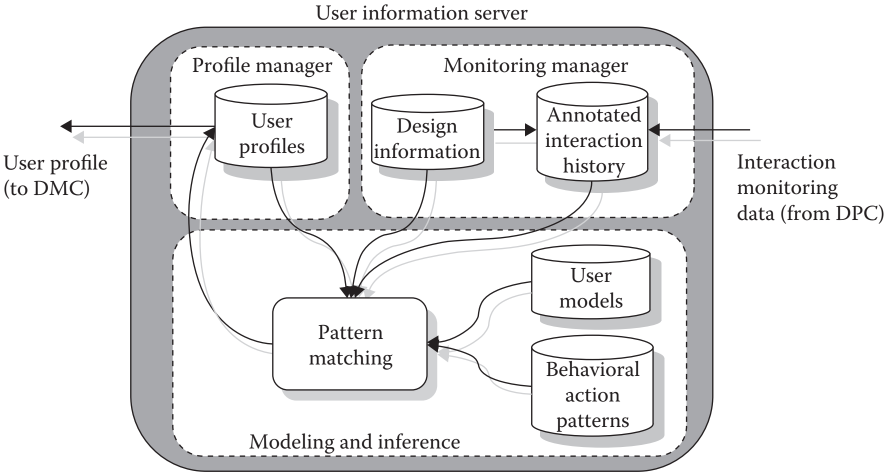


FIGURE 50.10 Internal microarchitecture of the user information server. User profile is posted to the decision-making component (DMC), while interaction-monitoring information is received from the dialogue patterns component (DPC).

Additionally, more sophisticated user representation and modeling methods can be used, including support for stereotypes of particular user categories. In case dynamic user-attribute detection is to be supported, the content may include dynamically collected interaction monitoring information, design information, and knowledge-processing components.

From a knowledge representation point of view, static or preexisting user knowledge may be encoded in any appropriate form; depending on the type of information, the user information server should feed to the decision-making process. Moreover, additional knowledge-based components may be used for processing retrieved user profiles, drawing assumptions about the user or updating the original user profiles. In Figure 50.10, the internal architecture of the user-information server used in the AVANTI web browser is presented. It should be noted that the first version of the AVANTI web browser produced in the context of the AVANTI Project used BGP-MS (Kobsa and Pohl 1995) for the role of the user-information server. The profile manager has been implemented as a database of profiles. The two other subsystems (e.g., monitoring manager, modeling, and inference) are needed only in case dynamic user-attribute detection is required. The interaction monitoring history has been implemented as a time-stamped list of monitoring events (the structure of monitoring events is described in the analysis of communication semantics) annotated with simple dialogue design context information (e.g., just the subtask name). In the user models, all types of dynamically detected user attributes have been identified (e.g., inability to perform a task, loss of orientation—those were actually the two dynamically detectable attributes required by the design in the AVANTI web browser). Each such attribute is associated with its corresponding behavioral action patterns. In the specific case, the representation of the behavioral patterns has been implemented together with the pattern-matching component, by means of state automata.

For instance, one heuristic pattern to detect loss of orientation has been defined as follows: The user moves the cursor inside the web-page display area, without selecting a link, for more than *N* seconds. The state automaton (see Figure 50.11) starts recording mouse moves in the page area, based on incoming monitored mouse moves, while finally triggering detection when no intermediate activity is successfully performed by the user. This worked fine from an implementation point of view. However, from the user interface design point of view, all such heuristic assumptions and behavioral patterns had to be extensively verified with real users so as to assert the relationship between the observable user behavior

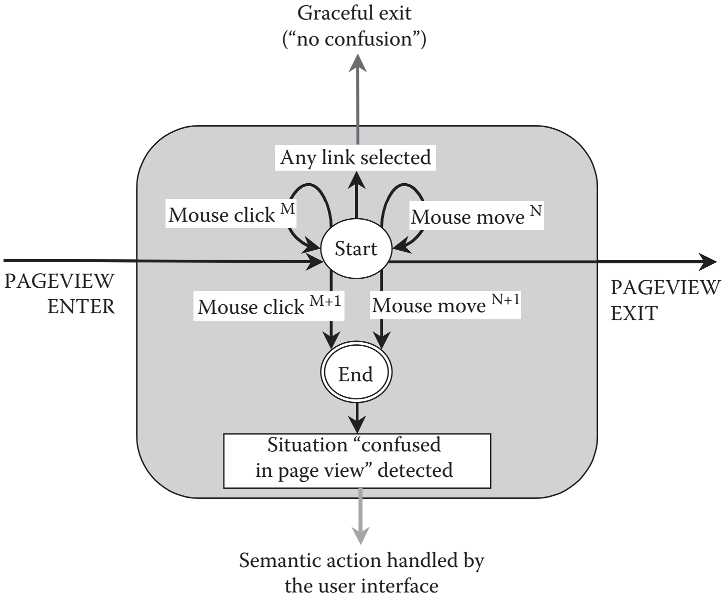


FIGURE 50.11 An example of an augmented state automaton used as an implementation technique for behavioral patterns; the state automaton is directly hard-coded in the UIS side, while multiple such automata coexist.

and the particular-inferred user attributes. This is a common issue in all adaptive systems that use heuristics for detecting user attributes at runtime, practically meaning that the validity of the assumptions inferred is dependent on the appropriateness of the specific user-action patterns chosen.

50.3.1.2 Context Parameters Server

The purpose of this component is to supply context attribute values (machine and environment) of two types: (1) (potentially) invariant, meaning unlikely to change during interaction, such as peripheral equipment and (2) variant, dynamically changing during interaction (e.g., environment noise, failure of particular equipment, etc.). This component is not intended to support device independence, but to provide device awareness. Its purpose is to enable the decision-making component to select those interaction patterns, which, apart from fitting the particular end-user attributes, are also appropriate for the type of equipment available on the end-user machine.

The usage-context attribute values are communicated to the decision-making component before the initiation of interaction. Additionally, during interaction, some dynamically changing usage-context parameters may also be fed to the decision-making component for decisions regarding adaptive behavior. For instance, assume that the initial decision for selecting feedback leads to the use of audio effects. Then, the dynamic detection of an increase in environmental noise may result in a runtime decision to switch to visual feedback (the underlying assumption being that such a decision does not conflict with other constraints).

This component encompasses a listing of the various invariant properties and equipment of the target machine (e.g., handheld binary switches, speech synthesizer for English, high-resolution display, mode 16 bits, 10243768, noisy environment, etc.). In this context, the more information regarding the characteristics of the target environment and machine is encapsulated, especially concerning input/output devices, the better adaptation can be achieved (information initially appearing redundant is likely to be used in future adaptation-oriented extensions).

The registry of environment properties and available equipment can be implemented easily as a profile manager in the form of a database. Such information will be communicated to the decision-making component as attribute/value pairs. However, if usage-context information is to be dynamically collected, such as environment noise or reduction of network bandwidth, the installation of proper hardware sensors or software monitors becomes mandatory.

50.3.1.3 Decision-Making Component

The role of this component is to decide, at runtime, the necessary adaptability and adaptivity actions and subsequently to communicate those to the dialogue patterns component (the latter being responsible for applying adaptation-oriented decisions). To decide adaptation, this component performs a kind of rule-based knowledge processing, to match end-user and usage-context attribute values to the corresponding dialogue artifacts, for all the various dialogue contexts.

This module encompasses the logic for deciding the necessary adaptation actions, on the basis of the user- and context-attribute values, received from the user-information server and the context parameters server, respectively. Such attribute values will be supplied to the decision-making component prior to the initiation of interaction within different dialogue contexts (e.g., initial values, resulting in initial interface adaptation), as well as during interaction (e.g., changes in particular values, or detection of new values, resulting in dynamic interface adaptations).

In the proposed approach, the encapsulated adaptation logic should reflect predefined decisions during the design stage. In other words, the inference mechanisms use well-defined decision patterns that have been validated during the design phase of the various alternative dialogue artifacts. In practice, this approach leads to a rule-based implementation, in which embedded knowledge reflects adaptation rules that have been already constructed and documented as part of the design stage. This decision-making policy is motivated by the assumption that if a human designer cannot decide upon adaptation for a dialogue context, given a particular end user and usage context, then a valid adaptation decision cannot be taken by a knowledge-based system at runtime. Later in this chapter, while discussing some implementation details of the AVANTI web browser, specific excerpts from the rule base of the decision engine will be discussed.

The first remark regarding the implementation of decision-making concerns the apparent awareness regarding (1) the various alternative dialogue artifacts (how they are named, e.g., virtual keyboard, for which dialogue context they have been designed, e.g., http address text field), and (2) user- and usage-context attribute names and their respective value domains (e.g., attribute age being integer in range 5 … 110).

The second issue concerns the input to the decision process, being individual user- and usage-context attribute values. Those are received at runtime both from the user-information server and from the context information server, either by request (e.g., the decision-making component takes the initiative to request the end-user and usage-context profile at start up to draw adaptability decisions) or by notification (e.g., when the user-information server draws assumptions regarding dynamic user attributes or when the context parameters server identifies dynamic context attributes).

The third issue concerns the format and structure of knowledge representation. In all developments that we have carried out, it has been proved that a rule-based logic implementation is practically adequate. Moreover, all interface designers engaged in the design process emphasized that this type of knowledge representation approach is far more close to their own way of rule-based thinking in deciding adaptation. This remark has led to excluding, at a very early stage, other possible approaches, such as heuristic pattern matching, weighting factor matrices, or probabilistic decision networks.

The final issue concerned the representation of the outcomes of the decision process in a form suitable for being communicated and easily interpreted by the dialogue patterns component. In this context, it has been practically proved that two categories of dialogue control actions suffice to communicate adaptation decisions: (1) activation of specific dialogue components and (2) cancellation of previously activated dialogue components. These two categories of adaptation actions provide the expressive power necessary for communicating the dialogue-component manipulation requirements that realize both adaptability and adaptivity. Substitution is modeled by a message containing a series of cancellation actions (e.g., the dialogue components to be substituted), followed by the necessary number of activation actions (e.g., which dialogue components to activate in place of the cancelled components). Therefore, the transmission of those commands in a single message (e.g., cancellation actions followed by activation actions) is to be used for implementing a substitution action. The need to send in one message packaged information regarding the cancelled component, together with the components that take its place, emerges when the implemented interface requires knowledge of all (or some of) the newly created components during interaction. For instance, if the new components include a container (e.g., a window object) with various embedded objects, and if upon the creation of the container information on the number and type of the particular contained objects is needed, it is necessary to ensure that all the relevant information (e.g., all engaged components) is received as a single message. It should be noted that, since each activation/cancellation command always carries its target UI component identification, it is possible to engage in substitution requests components that are not necessarily part of the same physical dialogue artifact. In addition, the decision to apply substitution is the responsibility of the decision-making component.

One issue regarding the expressive power of activation and cancellation decisions categories concerns the way dynamic interface updates (e.g., changing style or appearance, without closing or opening interface objects) can be effectively addressed. The answer to this question is related to the specific connotation attributed to the notion of a dialogue component. A dialogue component may not only implement physical dialogue context, such as a window and embedded objects, but may concern the activation of dialogue control policies or be realized as a particular sequence of interface manipulation actions. In this sense, the interface updates are to be collected in an appropriate dialogue implementation component (e.g., a program function, an object class, a library module) to be subsequently activated (e.g., called) when a corresponding activation message is received. This is the specific approach taken in the AVANTI web browser, which, from a software engineering point of view, enabled a better organization of the implementation modules around common design roles.

50.3.1.4 Dialogue Patterns Component

This component is responsible for supplying the software implementation of all the dialogue artifacts that have been identified in the design process. Such implemented components may vary from dialogue artifacts that are common across different user- and usage-context attribute values (e.g., no adaptation needed) to dialogue artifacts that will map to individual attribute values (e.g., alternative designs have been necessitated for adapted interaction). Additionally, as it has been previously mentioned, apart from implementing physical context, various components may implement dialogue-sequencing control, perform interface manipulation actions, maintain shared dialogue state logic, or apply interaction monitoring.

The dialogue patterns component should be capable of applying at runtime, activation, or cancellation decisions originated from the decision-making component. Additionally, interaction-monitoring components may need to be dynamically installed/uninstalled on particular physical dialogue components. This behavior will serve the runtime interaction monitoring control requests from the user-information server to provide continuous interaction monitoring information back to the user-information server for further intelligent processing. The dialogue patterns component either embeds the software implementation of the various dialogue components or is aware of where those components physically reside by used dynamic query, retrieval, and activation methods. The former is the typical method that can be used if the software implementation of the components is provided locally by means of software modules, libraries, or resident installed components. Usually, most of the implementation is to be carried out in a single programming language. The latter approach reflects the scenario in which distinct components are implemented on top of component-ware technologies, usually residing in local/remote component repositories (also called "registries" or "directories"), enabling reuse with dynamic deployment.

In the development of the AVANTI web browser, a combination of these two approaches has been used by implementing most of the common dialogue components into a single language (actually in C11 by using all the necessary toolkit libraries) while implementing some of the alternative dialogue artifacts as independent Active X components that were located and employed on the fly. The experience from the software development of the AVANTI web browser has proved that (1) the single language paradigm makes it far easier to perform quick implementation and testing of interface components and (2) the component-based approach largely promotes binary format reuse of implemented dialogue components while offering far better support for dynamic interface assembly, which is the central engineering concept of unified user interfaces (this issue will elaborate on the conclusion section of the chapter).

The microarchitecture of the dialogue patterns component internally used in the AVANTI web browser, as outlined in Figure 50.12, emphasizes internal organization to enable extensibility and evolution by adding new dialogue components. Additionally, it reflects the key role of the dialogue patterns component in applying adaptation decisions. The internal components are as follows.

The activation dispatcher locates the source of implementation of a component (or simply uses its application programming interfaces [APIs], if it is a locally used library) to activate it. In this sense, activation may imply a typical instantiation in OOP terms, calling of particular service functions or activating a remotely located object. After a component is activated, if cancellation is to be applied to this component, it is further registered in a local registry of activated components. In this registry, the indexing parameters used are the particular dialogue context (e.g., subtask, for instance, "http address field") and the artifact design descriptor (e.g., unique descriptive name provided during the design phase—for instance, "virtual keyboard"). For some categories of components, cancellation may not be defined during the design process, meaning there is no reason to register those at runtime for possible future cancellation (e.g., components with a temporal nature that perform only some interface update activities).

The cancellation dispatcher locates a component based on its indexing parameters and calls for cancellation. This may imply a typical destruction in OOP terms, calling internally particular service functions that may typically perform the unobtrusive removal of the physical view of the cancelled component or the release of a remote object instance. After cancellation is performed, the component instance is removed from the local registry.

The monitoring manager plays a twofold role: (1) It applies monitoring control requests originated from the user-information server by first locating the corresponding dialogue components and then requesting the installation (or uninstallation) of the particular monitoring policy (this requires implementation additions in dialogue components, for performing interaction monitoring and for activating or deactivating the interaction monitoring behavior) and (2) it receives interaction monitoring notifications from dialogue components and posts those to the user-information server.

The communication manager is responsible for dispatching incoming communication (activation, cancellation, and monitoring control) and posting outgoing communication (monitoring data and initial adaptation requests). One might observe an explicit link between the dialogue components and the communication manager. This reflects the initiation

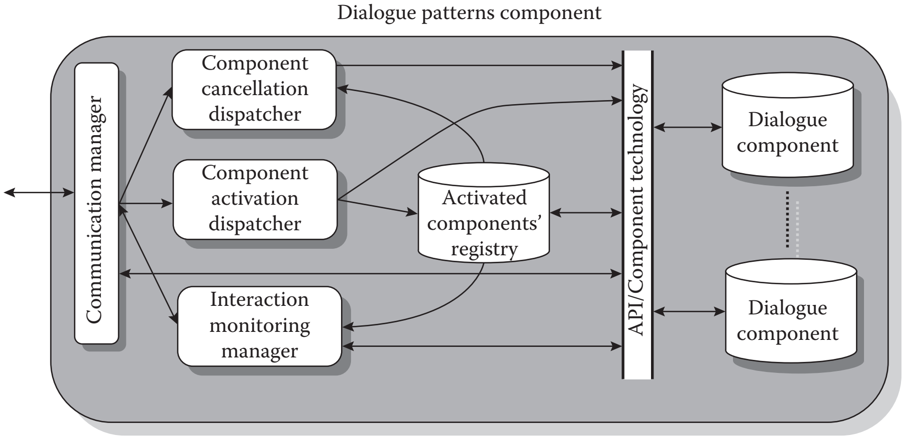

Dialogue patterns component
Communication manager
Component cancellation dispatcher
Component activation dispatcher
Activated components' registry
Interaction monitoring manager
API/Component technology
Dialogue component
Dialogue component

FIGURE 50.12 Internal microarchitecture of the dialogue patterns component.

of interaction in which the dialogue control logic (residing within dialogue components) requests iteratively the application of decision making (from the decision-making component). Such requests will need to be posted for all cases involving dialogue component alternatives for which adapted selection has to be appropriately performed.

The dialogue components typically encompass the real implementation of physical dialogues, dialogue control logic, and interaction monitoring method. In practice, it is hard to accomplish isolated implementation of the dialogue artifacts as independent black boxes that can be combined and assembled on the fly by independent controlling software. In most designs, it is common that physical dialogue artifacts are contained inside other physical artifacts. In this case, if there are alternative versions of the embedded artifacts, it turns out that to make containers fully orthogonal and independent with respect to the contained, one has to support intensive parameterization and pay a heavier implementation overhead. However, the gains are that the implementation of contained artifacts can be independently reused across different applications, while in the more monolithic approach, reuse requires deployment of the container code (and recursively, of its container too, if it is contained as well).

50.3.2 Adaptability and Adaptivity Cycles

The completion of an adaptation cycle, being either adaptability or adaptivity, is realized in a number of distributed processing stages performed by the various components of the unified architecture. During these stages, the components communicate with each other, requesting or delivering specific pieces of information. Figure 50.13 outlines the processing steps for performing both the initial adaptability cycle (to

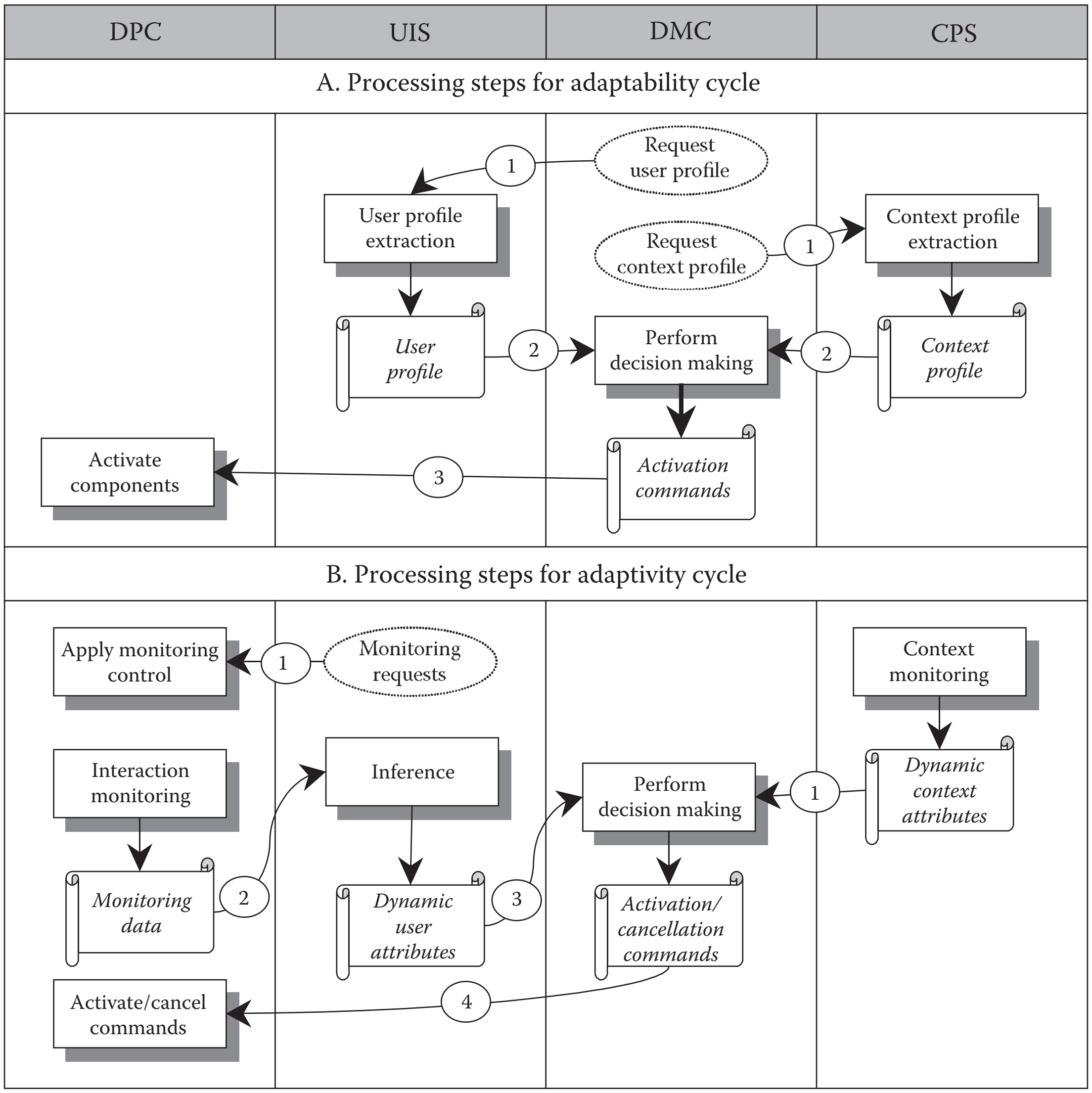


FIGURE 50.13 Processing steps, engaging communication among architectural components, to perform initial adaptability cycles (A), as well as the two types of adaptivity cycles (B). Requests originated from dialogue patterns component are indicated with dashed ovals, communicated messages are shown with special banners, processing points are drawn with shaded rectangles, while logical ordering is designated with numeric oval labels.

be executed only once) and the two types of adaptivity cycles (e.g., one starting from dynamic context attribute values and another starting from interaction monitoring control). Local actions indicated within components (in each of the four columns) are either outgoing messages, shown in bold typeface, or necessary internal processing, illustrated via shaded rectangles.

50.3.3 Dynamic User Interface Assembly

The concept of dynamic interface assembly reflects the key runtime mechanisms to support adaptability in unified user interfaces (Figure 50.14). Previous work in adaptive interaction involved techniques such as detection of user attributes, adaptive prompting, and localized lexical-level modifications (e.g., rearranging menu options, or adding/removing operation buttons). The issue of making the interface fit from the beginning to individual users has been addressed in the past mainly as a configuration problem, requiring interface developers to supply configuration editors so that end users could fit the interface to their particular preferences. However, such methods are limited to fine tuning some lexical-level aspects of the interface (e.g., toolbars, menus), while they always require explicit user intervention, for example, there is no automation. In this context, the notion of adaptability, as realized in unified user interfaces, offers new possibilities for automatically adapted interactions, while the architecture and runtime mechanisms to accomplish dynamic interface assembly constitute a unique software engineering perspective.

Some similarities with dynamic interface assembly can be found in typical web-based applications delivering dynamic content. The software engineering methods used in such cases are based on the construction of application templates (technologies such as Active Server Pages by Microsoft or Java Server Pages by JavaSoft are usually used) with embedded queries for dynamic information retrieval, delivering to the user a web page assembled on the fly. In this case, there are no alternative embedded components—just content to be dynamically retrieved—while the web-page assembly technique is mandatory when HTML-based web pages are to be delivered to the end user (in HTML, each time the content changes, a different HTML page has to be written). However, in case a full-fledged embedded component is developed (e.g., ActiveX object or Java Applet), no runtime assembly is required, since the embedded application internally manages content extraction and display, as a common desktop information retrieval application.

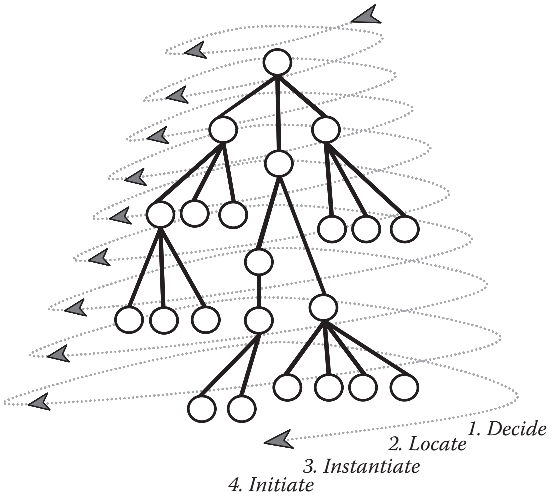


FIGURE 50.14 Illustration of the dynamic interface assembly process as an incremental hierarchical construction procedure.

The implementation of unified user interfaces is organized in hierarchically structured software templates, in which the key placeholders are parameterized container components. This hierarchical organization, as it has been reflected in the development excerpts, mirrors the fundamentally hierarchical constructional nature of interfaces. The ability to diversify and support alternatives in this hierarchy is due to containment parameterization, while the adapted assembly process is realized by selective activation, engaging remote decision making on the basis of end-user and usage-context information. The dynamic interface assembly process reflects the hierarchical traversal in the task hierarchy, starting from the root, to decide, locate, instantiate, and initiate appropriately every target user interface component (see Figure 50.15).

This process primarily concerns the interface components that implement alternative styles. From the implementation point of view, the following software design decisions have been made:

- The task hierarchy has been implemented as a tree data structure with polymorphic nodes triggering decision-making sessions (see Figure 50.15).
- Interface components have been implemented as distinct independent software modules, implementing generic containment APIs, while exposing a singleton control API for dynamic instantiation and name-based lookup.
- The interface assembly procedure is actually carried out via two successive hierarchical passes:
  - Execution of decision sessions, to identify the specific styles for polymorphic task contexts, which will be part of the eventually delivered user interface
  - Interface construction, through instantiation and initiation of all interface components for the decided styles

50.3.4 Polymorphic Containment Hierarchies

In Figure 50.16, the concept of parametric container hierarchies is illustrated, while in Figure 50.17, an instantiation of the concept is shown for the AVANTI web browser. Container classes expose their containment capabilities and the type of supported contained objects by defining abstract

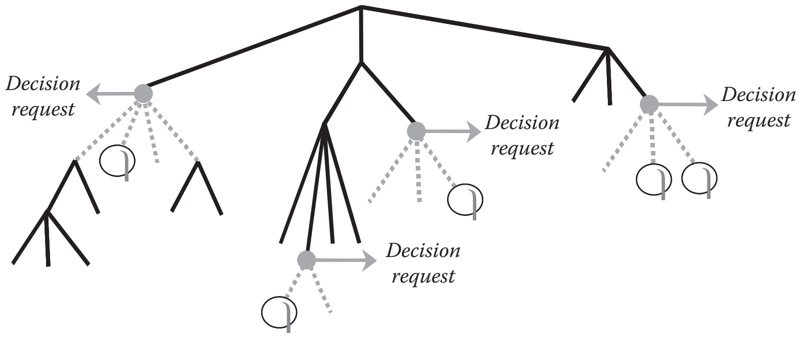


FIGURE 50.15 Illustrating the hierarchical posting of decision requests, causing decision sessions for each polymorphic task (shown with decomposition alternatives as dashed lines), and marking of selected alternative styles (i.e. interface components), after each decision session completes.

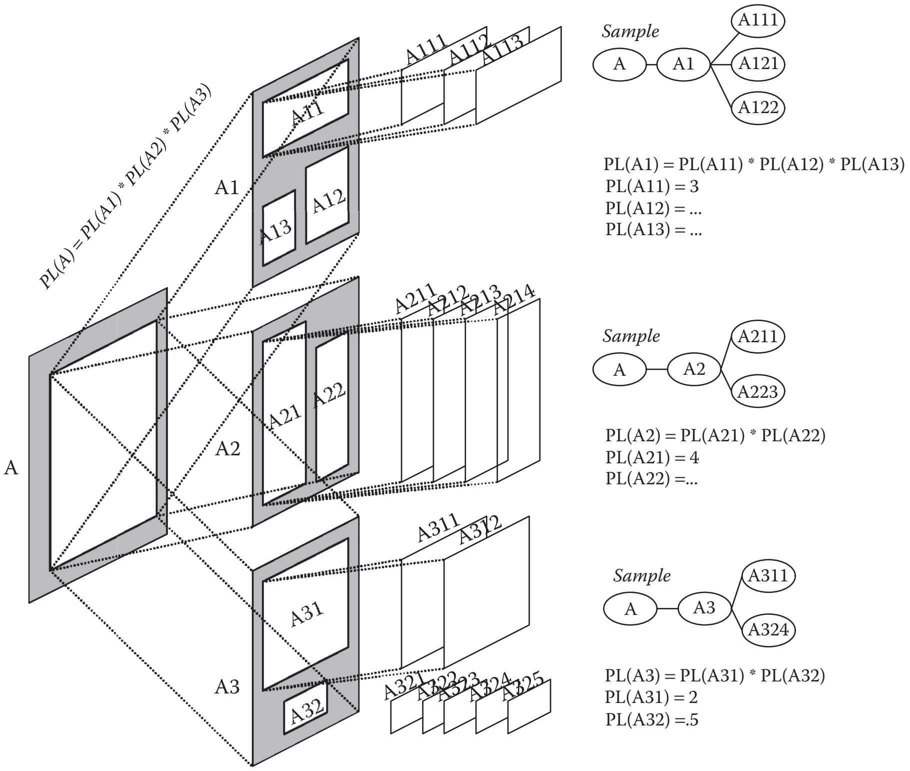


FIGURE 50.16 The structure of polymorphic containment hierarchies. Alternative contained components implement a common abstract interface that containers use and vice versa. PL indicates the maximum polymorphism factor, which provides the total number of all possible different runtime incarnations of an interface component, recursively defined as the product of the polymorphic factors of constituent component classes. Not all PL combinations may be semantically viable.

interfaces (e.g., abstract OOP classes) for all the contained component classes.

These interfaces, defined by container class developers, constitute the programming contract between the container and the contained classes. In this manner, alternative derived contained-component classes may be instantiated at runtime as constituent elements of a container. Following the definition of polymorphic factor PL, which provides a practical metric of the number of possible alternative runtime configurations of a component (not all of which may be semantically viable), the PL of the top-level application component gives the number of the possible alternative dynamically assembled interface instances. From a programming point of view, in the AVANTI web browser, the activation control of dialogue components for runtime assembly has been mainly realized through typical library function calls. Such function calls engage object instances corresponding to dialogue components, without using any component-ware technology. Hence, this runtime assembly behavior has been accomplished without the need of locating, fetching, and combining components together. Nevertheless, efforts have been devoted to applying and testing the latter approach in real practice, by using a component-ware technology (DCOM/ActiveX) for a limited number of dialogue components. This required a more labor-intensive implementation approach (from a C11 point of view, while isolated testing of components with VisualBasic was

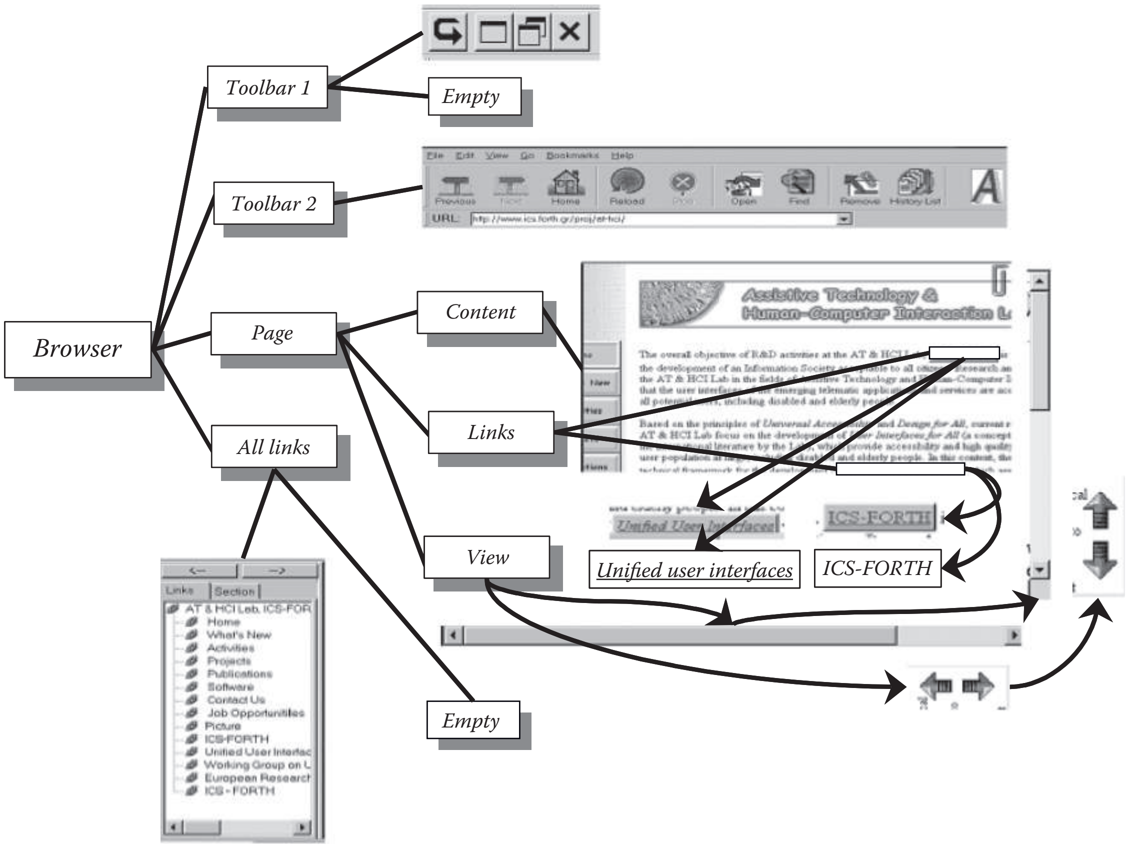


FIGURE 50.17 Parametric polymorphic containment with variant constituent components in the AVANTI browser. The indication "Empty" indicates components whose presence may have to be omitted upon dynamic interface delivery for certain user categories.

far easier) for packaging dialogues to make them component enabled, as well as for further activating and using them at runtime. However, there are some evident advantages:

- Dialogue components need not be carried altogether, but can be dynamically loaded, thus promoting a thin dialogue patterns component implementation.
- In effect, the core logic of the dialogue patterns component, apart from dialogue components, can be also packaged as a component itself, making it reusable across different applications.
- Automatic updates and extensions of components are directly supported, enabling new versions, or even new dialogue components (addressing more user- and usage-context attribute values) to be centrally installed in appropriate component repositories.

50.3.5 Decision-Making Specification

The decision-making logic is defined in independent decision blocks, each uniquely associated to a particular task context; at most one block per distinct task context may be supplied. The decision-making process is performed in independent sequential decision sessions, and each session is initiated by a request of the interface assembly module for execution of a particular initial decision block. In such a decision session, the evaluation of an arbitrary decision block may be performed, while the session completes once the computation exits from the initial decision block. The primary decision parameters are the end-user and usage-context profiles, defined as two built-in objects, such as user and context, whose attributes are syntactically accessible in the form of named attributes. The binding of attribute names to attribute values is always performed at runtime. The encapsulation of composite attributes in user and context profiles is easily allowed due to the syntactic flexibility of attribute reference. For instance, "user-abilities.vision" and "user-abilities.hearing" are syntactic sugar for "user-abilities.vision" and "user-abilities.hearing," where "abilities.vision" and "abilities.hearing" are two distinct independent ordinal attributes of the user built-in object. Consequently, even though all attributes in the decision-making specification language (DMSL) language are semantically scalar, the flexibility of attribute names allows syntactical simulation of aggregate structures.

In Figure 50.18, an example decision block is shown, being an excerpt of the implementation of the decision logic AVANTI web browser, for selecting the best alternative interface components for the link task context. The interface design relating to this adaptation decision logic is provided in Figure 50.1.

In the DMSL language, the fundamental design relationships among alternative styles are not injected as a part of the semantics but, alternatively, concrete rule patterns are delivered, effectively mapping those relationships

```
taskcontext link  [
       evaluate linktargeting;
       evaluate linkselection;
       evaluate loadconfirmation;
]

taskcontext linktargeting [
       if (user.abilities.pointing == accurate) then
              activate "manual pointing";
       else
              activate "gravity pointing";
]

taskcontext linkselection [
       if (user.webknowledge in {good, normal}) then
              activate "underlined text";
       else
              activate "push button";
]

taskcontext loadconfirmation [
       if (user.webknowledge in {low, none} or context.net==low) then
              activate "confirm dialogue";
       else
              activate "empty";
]
```

FIGURE 50.18 An example of a simple decision block to select the most appropriate delivery of web links for the individual end-user; notice that names in italics are not language keywords but are treated as string constants, that is, user.webknowledge is syntactic sugar for user."webknowledge."

Exclusion(S_1, S_2)	Compatibility(S_1, S_2)

if (S_1.*cond*) then
activate S_1;
else
if (S_2. *cond*) then
activate S_2;
	if (S_1.*cond*) then
	activate S_1;
	if (S_2. *cond*) then
	activate S_2;

Substitution(S_1 by S_2)	**Augmentation(S_1 by S_2)**

if (S_2. *cond and* isactive(S_1)) then [
cancelS1;
activate S2;
]
else
if (S_1.*cond*)
activate S_1;
	if (S_1.*cond*)
	if (not isactive(S_1))then
	activate S_1;
	else
	if (S_2. *cond*) then
	activate S_2;

FIGURE 50.19 The decision-rule patterns associated to the relationships among alternative styles; the style condition is the Boolean expression engaging the user and context attribute values for which the style is designed.

to implementation skeletons of decision blocks. This gives adaptation designers the freedom not to necessarily adopt those particular design relationships, in case, for instance, they do not choose to use unified design as the adaptation-design approach. In Figure 50.19, the DMSL decision-rule patterns are provided for the previously described four categories of style relationships.

50.4 SOFTWARE REFACTORING FOR ADAPTATION

We introduce a user-interface refactoring process to accommodate adaptive behavior into existing, even nonadaptive, user-interface implementations. Refactoring (Opdyke 1992), a term from mainstream software engineering, concerns the process of gradually applying small-scale source-level transformations (Mens and Tourwe 2004) aiming to enhance a system's software design, however, without affecting its domain-specific functionality. Refactoring is an incremental design improvement activity rather than a software reengineering process. Thus, refactoring is seen as a stepwise code evolution process to enhance an existing design, rather than to radically reform the software design itself. Radical reformations are handled by reengineering processes, being more resource demanding as they introduce severe architectural modifications.

Our approach, being a refactoring process, does not impose architectural refinements, meaning it respects the original user-interface software architecture. Instead, it emphasizes the introduction of targeted source code amendments, transforming gradually the class-level software design of user-interface components toward adaptive composition. The latter is accomplished because all transformations we adopt are existing refactoring patterns. Since refactoring is architecture-preserving refinement, so is our process. Technically, we treat adaptive user-interface composition and updating as a cross-cutting concern—sort of an aspect (Kiczales et al. 1997)—meaning it intersects with the software design at specific points, however, without affecting the global architectural picture.

Regarding adaptive user-interface behavior, there are various propositions all of which suggest customized architectures or processes for a development from scratch approach. For example, the proposal of a plasticity reference framework (Calvary et al. 2001) supports adaptation (retargeting) to different platforms. A model-driven approach for plasticity is proposed in Collignon et al. (2008). The adoption of a task-driven user-interface architecture for adaptive ambient intelligence is proposed in Clerckx et al. (2008), while adaptive information retrieval is addressed in Wen et al. (2007) as a way of adaptive content composition. As with the previous propositions, our work on unified user-interfaces also requires early adoption of our suggested user-interface architecture. All previous methods tend to be impractical for updating existing software systems to accommodate adaptive behavior for two primary reasons: (1) the adaptive behavior is imposed as the dominant view of the user-interface architecture, although the domain-specific noninteractive source code may be orders of magnitude larger compared to adaptivity-specific code and (2) they are quite diverse and their combined adoption in a single system is not investigated and might introduce practical issues due to differing architectural styles implied by varying goals: user intention extraction, dialogue automation, content adaptation, context adaptation, and cross-platform delivery.

Naturally, propositions that do not address user-interface engineering but focus on the reasoning to decide adaptation, such as Paternò et al. (2008), may be adopted once they do not require architectural refinements. In the context of our work, we investigated the underlying software engineering disciplines to support adaptive behavior, seeking for a common denominator amongst the alternative approaches. It quickly turned out that, irrespective of the eventual adaptive behavior, all methods entail three fundamental concepts:

- User-interface component alternatives
- Rationale runtime component selection
- Adaptive activation or replacement

Dynamic activation and replacement were very early recognized as the fundamental system actions to realize adaptive behavior (Cockton 1993). Thus, once supported, virtually any designed adaptation scenario is implementable. The complexity, size, and type of components widely vary, ranging from widgets to comprehensive dialogues, while adaptive activation may imply standalone presence, composition (aggregation), and replacement (substitution). In fact, these disciplines proved to be so fundamental that we could directly generalize from adaptive user-interfaces to adaptive software in general (Savidis 2004). However, even in this general proposition the practicality issue was not resolved: the original system architecture had to be always refined since a software reengineering process was suggested. At this point, our proposal for a refactoring process addresses this issue, showing that we can effectively enable adaptive behavior by treating adaptive composition as a cross-cutting concern that can be accommodated with well-defined incremental source-code transformations. We continue with an elaboration of this refactoring process.

50.4.1 Process Outline

We adopt a role as a responsibility-based notion for user-interface components, essentially abstracting over user-interface operations and requirements to denote functional requirements specific to roles. The latter reflects recent trends in software-design (Wirfs-Brock and Mc Kean 2003), where the emphasis is shifted from functionality-driven class-based design to responsibility-driven role-based design. We can have alternative implementations for a given role, role implementations being actual components, with *r*(*a*) denoting the role of user-interface component α. Following this, we define how relationships among components emerge by respective relationships among their actual roles:

Component *a*	⇒	*a* implements *r*(*a*)
Adaptive component *a*	⇒	*a* implements adaptive *r*(*a*)
a contains *b*	⇒	*r*(*a*) contains *r*(*b*)
a deploys *b*	⇒	*r*(*a*) deploys *r*(*b*)
a indifferent *b*	⇒	*r*(*a*) indifferent *r*(*b*)

It should be noted that contains and deploys relate to aggregation and deployment at the functional level, not the user-interface layout. The previous definitions state that component relationships are implied by the relationships of their abstract operations or roles. Implementation wise, roles map to components, and components map to concrete classes, modules, or packages. We rely on these two fundamental software relationships to drive structural transformations for all adaptively contained or deployed components.

Our overall software refactoring process is outlined under Figure 50.20, prescribing (1) three preparatory activities to extrapolate, model, and represent information from data already available at the end of the user-interface (re)design phase and (2) five concrete source code transformation activities to actually implement the adaptive behavior. We continue with an elaboration of the transformation phases, focusing on the preparatory phases not relating to source-code updates. The details of the user-interface source-code updates are provided under Savidis and Stephanidis (2010).

50.4.2 Identify Roles and Requirements

As mentioned earlier, adaptive composition involves at runtime the adaptation-driven selection and activation of

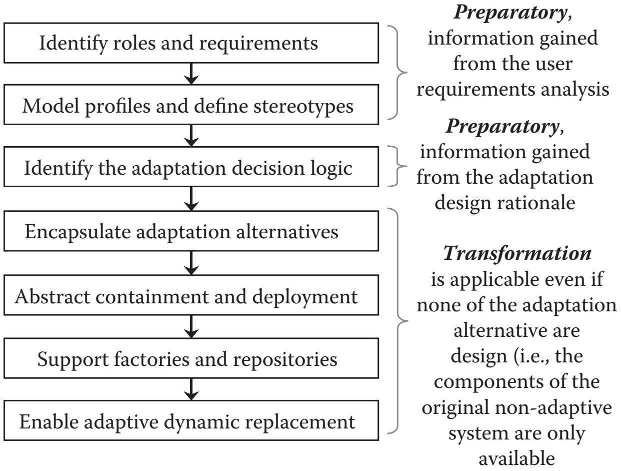

Identify roles and requirements
Model profiles and define stereotypes
Identify the adaptation decision logic
Encapsulate adaptation alternatives
Abstract containment and deployment
Support factories and repositories
Enable adaptive dynamic replacement
Preparatory, information gained from the user requirements analysis
Preparatory, information gained from the adaptation design rationale
Transformation is applicable even if none of the adaptation alternative are design (i.e., the components of the original non-adaptive system are only available

FIGURE 50.20 Adaptation-oriented user-interface refactoring process.

Roles	***Requirements***
Link	Allows activation of a target page
Scroller	Allows viewing of parts of large pages
Roles	Requirements
LinkTargeting	Allows targeting to a specific link

FIGURE 50.21 An example of user-interface component roles and respective requirements.

dialogue components, from a pool of related alternatives, with the design aim to optimally support respective user tasks. To refactor a system for adaptivity, we should initially associate concrete roles to all implemented user-interface components and identify their respective functional requirements. As an example, consider the roles and requirements defined in Figure 50.21 for the adaptive browser. What the specific list states is that Link is a user-interface role whose primary functional requirement is to facilitate activation of a target page. In case such brief information is not already part of the user-interface documentation, it can be easily extrapolated by reviewing the user-interface design outcomes or by referring to the designers.

50.4.3 Model Profiles and Define Stereotypes

Supporting adaptation means delivering appropriate variations of system behavior according to different deployment profiles. For interactive systems, such profiles concern user and usage contexts. Practically, the user-interface system exposes different user-interface profiles in response to deployment profiles. We use the term interface profile model to denote the model expressing the domain of variations of user-interface behavior. Such a model enumerates the viable alternatives for adaptive roles. For example, the model of Figure 50.22 relates to the adaptive browser (a specification pseudo language is used). This model designates the number of different adaptive composition possibilities in the browser, since every interface profile instance essentially represents a separate user-interface setup. Additionally, besides concrete roles like Links and Scroller, cross-cutting user-interface features, like Language, Audio Feedback, and so on, can be well expressed. Using this model, we can identify distinct setups as appropriate for sets of deployment profiles. For instance, let us consider the instantiations of our sample interface profile model, called stereotypes, also depicted under Figure 50.22.

Intuitively, there is a rationale link between the chosen stereotype names, like ForNaiveUsers, and the respective values given to the fields of the interface profile model. Such interface profile instances are called user-interface stereotypes and they document in a readable way key scenarios of adaptive setup, making more explicit the deployment profile accommodated with such setups. For example, one anticipates that the ForNaiveUsers stereotype is normally targeted to user profiles implying a “naïve end-user.” This step of distinguishing specific user-interface setups is optional. Its objective is to provide an initial understanding regarding the adaptivity potential of the original or refined user-interface implementation.

The next step is to define the deployment profile model representing information about the end-user and the usage context. User information may be stored in profile databases, may be gained from servers via unique user identifiers, can be extracted from a smart card, may be required user-input in a startup interaction session, or can be inferred at runtime from interaction monitoring and analysis. Similarly, context information, like location, environment noise, network bandwidth, and machine setup, may be provided using special-purpose equipment like sensors (for changing features) and a system-level profile (a registry for static features). Clearly, the definition of the deployment profile model should reflect information about the actual user population and the real environments of the use of the software system. An excerpt of the user profile model and a stereotype for the adaptive browser is provided under Figure 50.23.

```
interface profile model {
Link          : { NoCancellation, WithCancellation }
Scroller             : { OutsidePage, InsidePage }
AllLinks             : { Supported, Unsupported }
LinkTargeting : { Manual, Assisted }
}
interface stereotype ForNaiveUsers {
 Link         = WithCancellation
Scroller             = OutsidePage
AllLinks             = Unsupported
LinkTargeting        = Assisted
}
interface stereotype ForExpertUsers {
 Link         = NoCancellation
Scroller             = InsidePage
AllLinks             = Supported
LinkTargeting = Manual
}
```

FIGURE 50.22 An example of the user-interface profile model with two stereotypes.

```
user profile model {
ComputerLiteracy            : { Good, Average, Some, None }
WebUse                      : { Frequent, Casual, None }
UserAge                     : { 4 .. 90 }
NativeLanguage              : Language
LanguagesSpoken             : list of Language
}
user stereotype Naïve {
    ComputerLiteracy        in { Some,   None}      or
    WebUse                  in {Casual, None}
}
```

FIGURE 50.23 An example of the user profile model with one stereotype.

It should be noted that during this phase the focal point is not on user modeling, but on profile modeling. The latter is technically straightforward, with complexity analogy to definition of database records, aiming to identify the structure of end-user records. The former is a rather complicated activity, entailing the modeling of user populations for reasoning or inference purposes, and is not required as such in our proposed refactoring process. However, the latter is not restrictive, in the sense that user-interface developers may well incorporate user models and reasoning components to support various intelligent user-interface functions.

50.4.4 Identify the Adaptation Decision Logic

Typically, during an adaptation-oriented (re)design phase, the concrete design rationale of every adaptive user-interface component is defined and documented. Such rationale encompasses the actual conditions regulating the delivery and presence of an adaptive component during interaction (runtime). In other words, it defines when adaptive activation of components should be performed and which of the available alternatives of adaptive components should be chosen. Only once such information is available the design phase gracefully concludes. Conceptually, such rationale constitutes a form of decision logic for adaptive component activation or deactivation of user-interface components. Additionally, such logic rationally links the deployment profile with designed user-interface components, meaning it is user and context dependent. Linking with our recent definitions, the adaptation decision logic links directly deployment profile attributes to interface profile attributes. As a trivial example, the end-user native language is typically used to choose the user-interface language. Examples of adaptation rules expressed in our Decision-DMSL (Savidis and Stephanidis 2005) supporting declarative rules with an imperative syntax are provided under Figure 50.24; notice the use of stereotypes in decision conditions (Link being an adaptive component).

Practically, developers may collect and implement the decision logic using any convenient method, including hard-coding in the user-interface implementation language, while express all profiles directly in XML. The most common approach is to introduce a separate special-purpose class responsible for decision making during runtime. Later, once the entire adaptation logic is consolidated and implemented and the system is transformed to an adaptive one, alternative implementation techniques may be explored better fitting the notion of a decision-making module (like rule-based systems or logic programming methods). Because of the simplicity of this implementation task and because it does not affect the existing user-interface system, we consider it more of a preparatory action rather than a transformation step. Summing up, the job of this phase is the collection and formulation

```
component Link {
      if   Naïve then            activate WithCancellation
      else                       activate NoCancellation
}
component LinkTargeting {
      if   Naïve or Elderly then activate Assisted
      else                       activate Manual
}
```

FIGURE 50.24 Sample adaptation rules expressed in decision-making specification language.

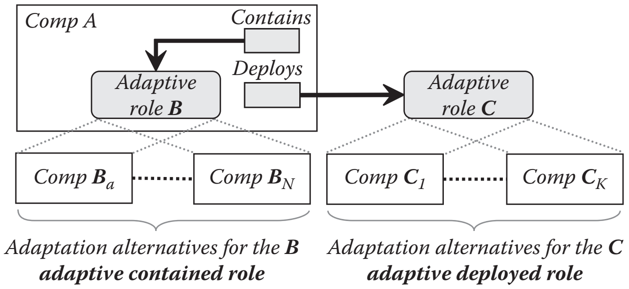

Comp A
Contains
Adaptive role B
Deploys
Adaptive role C
Comp B_a
Comp B_N
Comp C_1
Comp C_K
Adaptation alternatives for the B adaptive contained role
Adaptation alternatives for the C adaptive deployed role

FIGURE 50.25 The component-based notion of adaptive polymorphic containment and deployment for applications encompassing adaptive components.

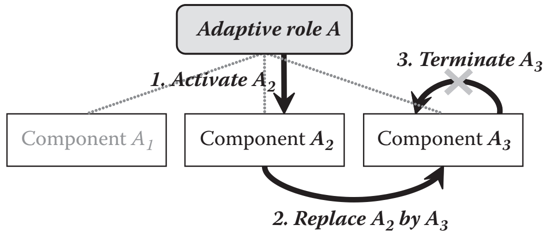


FIGURE 50.26 The component-based notion of adaptive dynamic replacement for applications encompassing adaptive components.

of the adaptation design rationale in a more formal style, being closer to a logic or algorithm form, that is, a computable representation. Clearly, this preparatory phase reflects an intention to make interactive systems capable to execute adaptation-related logic so as to realize a runtime decision making for required adaptation actions.

50.4.5 Apply Source Code Transformations

The application of source code transformation steps aims to gradually bring the implementation into a form where adaptive *composition* and *replacement* are fully supported as key functional features. Architecturally, these two disciplines will affect only microscopically the system design, down to the level of component interactions and dependencies, as outlined in Figures 50.25 and 50.26. Technically, the refactoring process will bring the source code into an eventual state where all adaptation-specific user-interface components may be essentially plugged in well-defined points, in a modular and extensible manner. As mentioned, the details of the source transformation steps are provided in Savidis and Stephanidis (in press).

50.5 CONCLUSIONS

Currently, the required technical knowledge to build user- and usage-context adapted interfaces includes user modeling, task design, cognitive psychology, rule-based systems, network communication and protocols, multiplatform interfaces, component repositories, development tools, and core user interface software engineering. Software development firms apparently prefer incremental engagement strategies, allowing a stepwise entrance to new potential markets, by delivering successive generations of products encompassing layers of novel characteristics. Similarly, the development of software applications supporting automatic user interface adaptation, for the broadest end-user population, requires a concrete strategy supporting evolutionary development, software reuse, incremental design, scalability, and modular construction. The unified user interface development discussed claims to offer a software engineering proposition that consolidates process-oriented wisdom for constructing automatically adapted user interfaces. Evolution, incremental development, and software reuse are some of the fundamental features of unified user interface development. These are reflected in the ability to progressively extend a unified user interface by incrementally encapsulating computable content in the different parts of the architecture, to cater for additional users and usage contexts, by designing and implementing more dialogue artifacts, and by embedding new rules for the decision-making logic. Such characteristics are particularly important and relevant to the claimed feasibility and viability of the proposed software engineering process and directly facilitate the practical accomplishment of universally accessible interactions.

The concept of unified user interfaces reflects a new software engineering paradigm that addresses effectively the need for interactions automatically adapted to the individual end-user requirements and the particular context of use. Following this technical approach, interactive software applications encompass the capability appropriately to deliver on the fly an adapted interface instance, performing appropriate runtime processing that engages the following:

- Utilization of user- and usage-context-oriented information (e.g., profiles), as well as the ability to detect dynamically user- and usage-context attributes during interaction.

- Management of appropriate alternative implemented dialogue components, realizing alternative ways for physical-level interaction.
- Adaptation-oriented decision making that facilitates (a) the selection, before initiation of interaction, of the most appropriate dialogue components comprising the delivered interface, given any particular dialogue context, for the particular end-user and usage-context profiles (e.g., adaptability) and (b) the implementation of appropriate changes in the initially delivered interface instance, according to dynamically detected user- and usage-context attributes (e.g., adaptivity).
- Runtime component co-ordination and control to dynamically assemble or alter the target interface; this user interface is composed on the fly from the set of dynamically selected constituent dialogue components.

The unified user interface development strategy provides a distributed software architecture with well-defined functional roles (e.g., which component does what), intercommunication semantics (e.g., which component requests what and from whom), control flow (e.g., when to do what), and internal decomposition (e.g., how the implementation of each component is internally structured). One of the unique features of this development paradigm is the emphasis on dynamic interface assembly for adapted interface delivery, reflecting a software engineering practice with repository-oriented component organization, parametric containers with abstract containment APIs, and common interaction-monitoring control with abstract APIs. Although the method itself is not intended to be intensively prescriptive from the low-level implementation point of view, specific successful practices that have been technically validated in fieldwork regarding decision making and dynamic user-attribute detection have also been discussed, focusing on microarchitecture details and internal functional decomposition.

In this context, this development method has been systematically deployed and tested in practical situations where automatic personalization of computer-based interactive applications and services was the predominant issue. It introduces the fundamental notion of adapted interface delivery before initiation of interaction and addresses the technical challenges of coping with the inherent runtime dynamic interface assembly process.

Additionally, the basic software engineering strategy of unified development, in particular abstract objects and polymorphic containment, has been deployed in leading-edge developments, to deliver applications with ambient mobile user interfaces running on wearable pocketsize processing units.

The proposed approach establishes one possible technical route toward constructing automatically individualized best fit user interfaces: it enables incremental development and facilitates the expansion and upgrade of dialogue components as an on-going process, entailing the continuous engagement and consideration of new design parameters, and new parameter values. It is anticipated that future research work may reveal alternative approaches or methods. At the same time, further research and development work for unified user interfaces is required to address some existing challenges, mainly related to design issues (see Figure 50.27).

Following Figure 50.25, one top-level issue concerns the way that specific varying user attributes affecting interaction

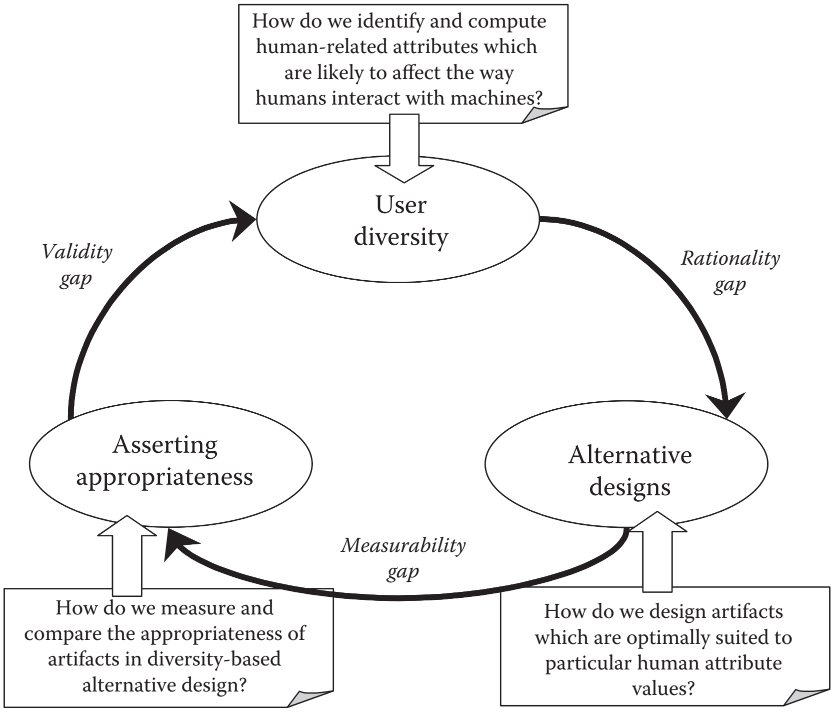

How do we identify and compute human-related attributes which are likely to affect the way humans interact with machines?
User diversity
Validity gap
Rationality gap
Asserting appropriateness
Alternative designs
Measurability gap
How do we measure and compare the appropriateness of artifacts in diversity-based alternative design?
How do we design artifacts which are optimally suited to particular human attribute values?

FIGURE 50.27 Open research questions in design for all.

are to be identified. In other words, there is a need to identify diversity in those human characteristics that are likely to dictate alternative dialogue means. Subsequently, even if a set of those attributes is identified, it is still unclear how to conduct a design process to produce the necessary alternative dialogue artifacts for the different values of those attributes. Hence, it is necessary to design for diversity relying upon appropriate design rationale clearly relating diverse attribute values with specific properties of the target dialogue artifacts. Currently, there is only limited knowledge about how to perform effectively this transition from alternative user-attribute values to alternative design artifacts, and it can be characterized as a rationalization gap.

The issue of how to structure appropriately alternative patterns for diverse user-attribute values should be addressed in a way that the resulting designs are indeed efficient, effective, and satisfactory for their intended users and usage contexts. Such a process requires appropriate evaluation methods and the capability to measure the appropriateness of designed artifacts. At present, this is still a missing link, characterized as the measurability gap. Unless we are able to assert the appropriateness of the alternative dialogue artifacts designed for diverse user attributes, we cannot validate the overall dynamically delivered interface. The inability to formulate and conduct such an evaluation process creates a validity gap. Work currently underway, as well as future work, is expected to address these issues in an attempt to bridge the identified gaps.

Finally, we outlined a software refactoring process to support adaptive user-interface composition and replacement for systems not originally designed to support such adaptive behavior. Our work is motivated by the fact that, while adaptivity gains broad interest for software products and services, all known propositions imply development from scratch and adoption of architectural styles that may not necessarily interoperate with the domain-specific software architecture. Our focus on software refactoring rather than on software reengineering is fundamental. More specifically, via reengineering, we need a process to fuse two parallel system designs and architectures together: the original domain-specific system design and architecture and the one implied by the need for adaptive behavior. Not only do we lack today such processes for interactive systems, but we lack software reengineering processes for software systems in general. By adopting a refactoring process, we have the extra key benefit that after every transformation activity the system is always in a fully working state. Overall, we believe that analogous refactoring processes for different categories of demanding user-interface features may lead to their easier adoption in real production systems.

ACKNOWLEDGMENTS

Recent work regarding the software refactoring process for adaptive user-interface behavior has been carried out in the context of the Ambient Intelligence Programme (AmI) of ICS-FORTH (http://www.ics.forth.gr/ami/index.html). This effort has been motivated by the observation that during the development of diverse and demanding applications in an AmI environment, it was impractical to simultaneously focus on the delivery of user-interface adaptation and radically new interaction metaphors. Thus, we needed an effective and efficient software transformation process to retrofit any AmI application to adaptation demands emerging after development.

REFERENCES

Calvary, G., J. Coutaz, and D. Thevenin. 2001. A unifying reference framework for the development of plastic user interfaces. In *Proceedings of EHCI2001 Conference*, 173–92. Berlin: Springer LNCS 2254.

Clerckx, T., C. Vandervelpen, K. Luyten, and K. Coninx. 2008. A task-driven user-interface architecture for ambient intelligent environments. In *Proceedings of the IUI 2008 Conference*, 309–11. New York: ACM.

Cockton, G. 1993. Spaces and distances—Software architecture and abstraction and their relation to adaptation. In *Adaptive User Interfaces—Principles and Practice*, 79–108. Amsterdam: Elsevier Science.

Collignon, B., J. Vanderdonckt, and G. Calvary. 2008. Model-driven engineering of multi-target plastic user interfaces. In *Proceedings of the Fourth International Conference on Autonomic and Autonomous Systems, ICAS '08*, ed. D. Greenwood, M. Grottke, H. Lutfiyya and M. Popescu, 16–21. March 2008, Gosier, Guadeloupe. Los Alamitos, CA: IEEE Computer Society. 2008, ISBN 978-0-7695-3093-2.

Hartson, H. R., A. C. Siochi, and D. Hix. 1990. The UAN: A user-oriented representation for direct manipulation interface design. *ACM Trans Inf Syst* 8(3):181–203.

Hill, R. 1986. Supporting concurrency, communication and synchronisation in human-computer interaction—The Sassafras UIMS. *ACM Trans Graph* 5(3):289–320.

Hoare, C. A. R. 1978. Communicating sequential processes. *Commun ACM* 21(8):666–77.

Johnson, P., H. Johnson, P. Waddington, and A. Shouls. 1988. Task-related knowledge structures: Analysis, modeling, and applications. In *People and Computers: From Research to Implementation—Proceedings of HCI '88*, ed. D. M. Jones and R. Winder, 35–62. Cambridge, MA: Cambridge University Press.

Kiczales, G., J. Lamping, A. Mendhekar, C. Maeda, C. Lopes, J. M. Loingtier, and J. Irwin. 1997. Aspect-oriented programming. In *Proceedings of the European Conference on Object-Oriented Programming*, 1241:220–42. Berlin: Springer LNCS.

Kobsa, A., and W. Pohl. 1995. The user modelling shell system BGP-MS. *User Modell User-Adapted Interact* 4(2):59–106.

Marcus, A. 1996. Icon design and symbol design issues for graphical interfaces. In *International User Interfaces*, ed. E. Del Galdo and J. Nielsen, 257–70. New York: John Wiley and Sons.

Mens, T., and T. Tourwe. 2004. A survey of software refactoring. *IEEE Trans Softw Eng* 30(2):126–39.

Opdyke, W. 1992. *Refactoring: A Program Restructuring Aid in Designing Object-Oriented Application Frameworks*. Ph.D. thesis, University of Illinois at Urbana-Champaign.

Saldarini, R. 1989. Analysis and design of business information systems. In *Structured Systems Analysis*, 22–3. New York: MacMillan Publishing.

Savidis, A. 2004. Dynamic software assembly for automatic Deployment-Oriented adaptation. *Elsevier Electron Notes Theor Comput Sci (ENTCS)* 127(3):207–17.

Savidis, A., C. Stephanidis, and D. Akoumianakis. 1997. Unifying toolkit programming layers: A multi-purpose toolkit integration module. In *Proceedings of the 4th Eurographics Workshop on Design, Specification and Verification of Interactive Systems (DSV-IS '97)*, Granada, Spain, June 4–6, ed. M. D. Harrison and J. C. Torres, 177–92. Berlin: Springer-Verlag.

Savidis, A., and C. Stephanidis. 2004. Unified user interface design: Designing universally accessible interactions. *Int J Interact Comput* 16:243–70.

Savidis, A., and C. Stephanidis. 2005. Distributed interface bits: Dynamic dialogue composition from ambient computing resources. *ACM-Springer J Pers Ubiquitous Comput* 9(3):142–68.

Savidis, A., and C. Stephanidis. 2010. Software refactoring process for adaptive user-interface composition. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS 2010)*, 19–23 June 2010, Berlin, Germany, 19–28. New York: ACM Press.

Stephanidis, C., A. Paramythis, M. Sfyrakis, and A. Savidis. 2001. A case study in unified user interface development: The AVANTI Web Browser. In *User Interfaces for All*, ed. C. Stephandis, 525–68. Mahwah, NJ: Lawrence Erlbaum Associates.

Wen, Z., M. Zhou, and V. Aggarwal. 2007. Context-aware adaptive information retrieval for investigative tasks. In *IUI 2008*, 122–31. New York: ACM.

Wirfs-Brock, R., and A. Mc Kean. 2003. Object Design: Roles, Responsibilities, and Collaborations. Boston, MA: Addison-Wesley.

51 Usability + Persuasiveness + Graphic Design = eCommerce User Experience

Deborah J. Mayhew

CONTENTS

51.1 DEFINING eCOMMERCE USER EXPERIENCE

In 1995, most websites were informational rather than transactional, the dot-com boom was in its infancy, and the web development world was only just beginning to rub shoulders with the longstanding field of software usability engineering. That year, Amazon and eBay were launched and Yahoo! was incorporated.

By 2000, the first U.S. dot-coms started to go out of business. But in spite of the bubble bust, over the past 10 years, eCommerce has become ubiquitous on the web, and the field of software usability has become much more visible and active in the web development world. As the web has matured with respect to usability, the field of traditional software usability (which dates back to the late 1970s) has come to recognize—and integrate with—other qualities of what is now referred to as the web "user experience." As web capabilities increased, *graphic design* has become a key quality of the user experience. And in the case of eCommerce websites, a relatively new quality of the user experience design has emerged: *persuasiveness*. At this point, any eCommerce designer or developer needs to recognize the importance of five different qualities of the user experience:

1. Utility
2. Functional integrity
3. Usability
4. Persuasiveness
5. Graphic design

These are defined in Sections 51.1.1 through 51.1.5.

51.1.1 UTILITY

It is easy to overlook *utility* as a quality of a website's user experience, as it is perhaps the most fundamental. The utility of a website refers to the usefulness, importance, or interest of the site content (i.e., of the information, products, or services offered by the site) to the visitor. It is of course relative to any particular site visitor—what is interesting or useful to you may not be to me. It is also a continuous quality, that is, some websites will feel more or less useful or interesting to me than others. For example, many web users love to use social networking sites like YouTube or Facebook, while others find these a total waste of time. I will have no need for a website that sells carpenter's tools, while my neighbor might visit and use that site on a regular basis.

51.1.2 FUNCTIONAL INTEGRITY

A website's *functional integrity* is simply the extent to which it works as intended. Websites may have "dead" links that go nowhere, they may freeze or crash when certain operations are invoked, they may display incorrectly on some browsers or browser versions, they may download unintended files, and so on. A lack of functional integrity is the symptom of buggy or incorrect code. Functional integrity is a continuous quality—some websites may only have a few insignificant bugs, others may be almost nonfunctional, and anything in between is possible.

51.1.3 Usability

Usability of course refers to how easy to learn (for first time and infrequent visitors) and/or use (for frequent visitors) a website is. A site can have high utility and high functional integrity and still be very difficult to learn or inefficient and tedious to use. For example, the website you use to submit your tax returns may be implemented in flawless code and be relevant to almost every adult, with great potential for convenience and cost savings, but be terribly hard to learn or inefficient to use. Conversely, a site can be very usable, but not very useful, or have low functional integrity. It might be very easy and intuitive to figure out how to perform a task, but the site may consistently crash at a certain point in the task flow so that the task can never be accomplished.

51.1.4 Persuasiveness

Utility, functional integrity, and usability are qualities important to virtually any website based on any underlying business model. When we focus on eCommerce sites in particular, another quality—*persuasiveness*—becomes very important.

Persuasiveness refers to the extent to which the user interface of a website encourages and promotes "conversions." What constitutes a conversion varies from site to site, and even non-eCommerce sites may be promoting some type of conversion (e.g., newsletter signup, switching to online tax filing). But persuasiveness is a particularly important user experience quality on an eCommerce site, and the primary type of conversion in this case is a sale. So in the case of eCommerce sites, persuasiveness refers mainly to the extent to which the user experience encourages and promotes sales.

Two key aspects of the quality of persuasiveness involve the presence and location of two types of information: *vendor* information (e.g., company name, physical address and contact information, company history, testimonials of past customers, etc.) and *product* information (things like product color, material, care instructions, etc.). Visitors look for evidence that they can trust an online vendor, especially if they have never heard of them before. And they are often unwilling to order a product if they do not know everything they need to know to judge whether it will meet their needs. This is why many people will often look for a product on Amazon.com first—because it is a trusted vendor, and it usually provides comprehensive product information, including detailed reviews by other customers. These two types of information are key to persuasion on eCommerce websites. And note that a website can be fully functional, highly usable in terms of task completion, and offer just what a visitor is looking for—but if it lacks key aspects of persuasiveness such as adequate vendor and product information, potential sales may be lost.

If, for example, I can easily find an attractive suit on an apparel site and easily check out, but I cannot tell if the suit requires dry cleaning, I will probably not order it. Similarly, if I cannot tell what the shipping charges will be before entering my credit card number, I may not order it. It is not that I cannot figure out how to complete the purchase process (usability), nor that I am put off by the look and feel of the website (graphic design), nor that the site crashes during the checkout process (functional integrity), nor that I cannot find a product I want (utility). What happens in these examples is that the website fails to give me adequate product information, or fails to keep me engaged in the purchase process by failing to give me the information I need (shipping costs) when I need it to make my buy decision.

51.1.5 Graphic Design

Finally, the "look and feel"—that is the *graphic design*—of a website can be a key part of the user experience. The graphic design of a website—primarily the way colors, images, and other media are used—invoke emotional reactions in visitors that may or may not contribute to the site's goals. A website's graphic design may strike a visitor as appealing, entertaining, or pleasing, or it may impress them as unprofessional, boring, or even offensive. As with other user experience qualities, each visitor's reaction to a given graphic design may be different. You may be bored by soft pastel colors while I may feel reassured and calmed by them. You may find a straightforward and simple graphic design boring while to me it may feel professional and reassuring. I may be put off by sound and animation, while you may find it exciting and appealing.

While utility and functional integrity are fairly independent qualities, the lines between usability, persuasiveness, and graphic design are more blurred. Clearly, usability and effective graphic design will contribute to persuasiveness, and graphic design can contribute significantly to usability. Nevertheless it is useful to consider these qualities separately to understand their importance and apply them effectively during design.

51.2 ACHIEVING A GREAT USER EXPERIENCE

On eCommerce websites, a great user experience is achieved by optimizing each of the user experience qualities defined earlier, relative to the intended market. Whole professions have evolved around each of these qualities.

The prerequisite of a great eCommerce website user experience is of course *utility*. Nothing else will help if a site does not offer anything of use or interest to a given visitor or market. Website businesses that do not do the research to determine the viability or competitiveness of particular products or services will not succeed regardless of other qualities of their site design. The age-old profession of *market research* is the relevant discipline to use here. Potential web-based businesses need to establish that they have a product or service that there is a market for and that they can compete with current vendors effectively.

Clearly, web businesses must insure that in the end, before launch, their website is comprehensively debugged and works without problems on at least the major browsers/browser versions used by their intended market. Nothing is more frustrating, and feels more unprofessional, than a website that breaks down. Visitors are not likely to return. Competent *web development* professionals are necessary to ensure *functional integrity*.

eCommerce websites need to be intuitive or at least easy to learn for first time and infrequent visitors, and if a website has them, efficient and easy to use for power visitors. *Software and web usability engineering* is the expertise needed to achieve the quality of *usability* in eCommerce user experience design.

eCommerce websites need to provide all critical information to support visitor decision making around their needs and desires and to provide it at the right time in the conversion task flow. There is a currently small but growing field of experts with experience applying marketing and *persuasion psychology* to eCommerce web design.

Finally, an eCommerce website needs a graphic design that inspires trust and is appealing and motivating to its intended market. *Graphic design* professionals specializing in website design provide the design skills and expertise in branding that eCommerce businesses need.

The real key here, beyond simply finding resources with the above skill sets, is to build an effective interdisciplinary design team. Often, professionals with these different backgrounds and skill sets are unfamiliar with the other disciplines and how they must work together to achieve an optimal user experience design for a given market. At the very least, eCommerce businesses need team members respectful of the expertise of others and with a willingness to learn to collaborate effectively to achieve the common goal of an optimized user experience design.

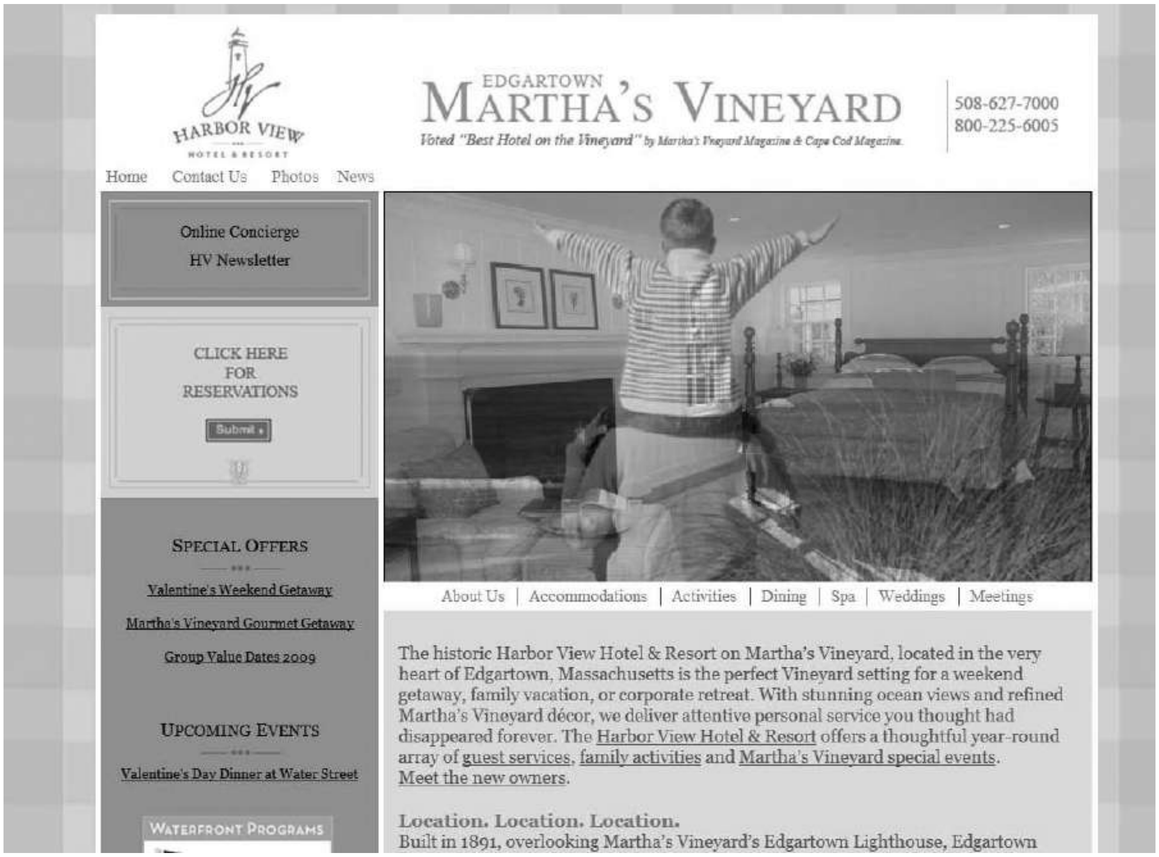
To make the differences between the different user experience qualities—and the disciplines behind them—more concrete, let us look at some existing eCommerce website pages with these qualities in mind. We will focus on the three qualities that are most visibly part of the user experience:

1. Usability
2. Persuasiveness
3. Graphic design

51.2.1 Usability

Figure 51.1 shows the home page of a hotel site (http://www.harbor-view.com/). At the top you may notice that the large image area is in the process of fading out of one photograph into another. This area offers an automatic slide show of lovely and appealing images of the hotel and surrounding areas. In fact, if visitors would like to, they can take control of this slide show by clicking anywhere in the image area; then the automatic slide show ceases and subsequent clicks will move through the images at the visitor's preferred pace.

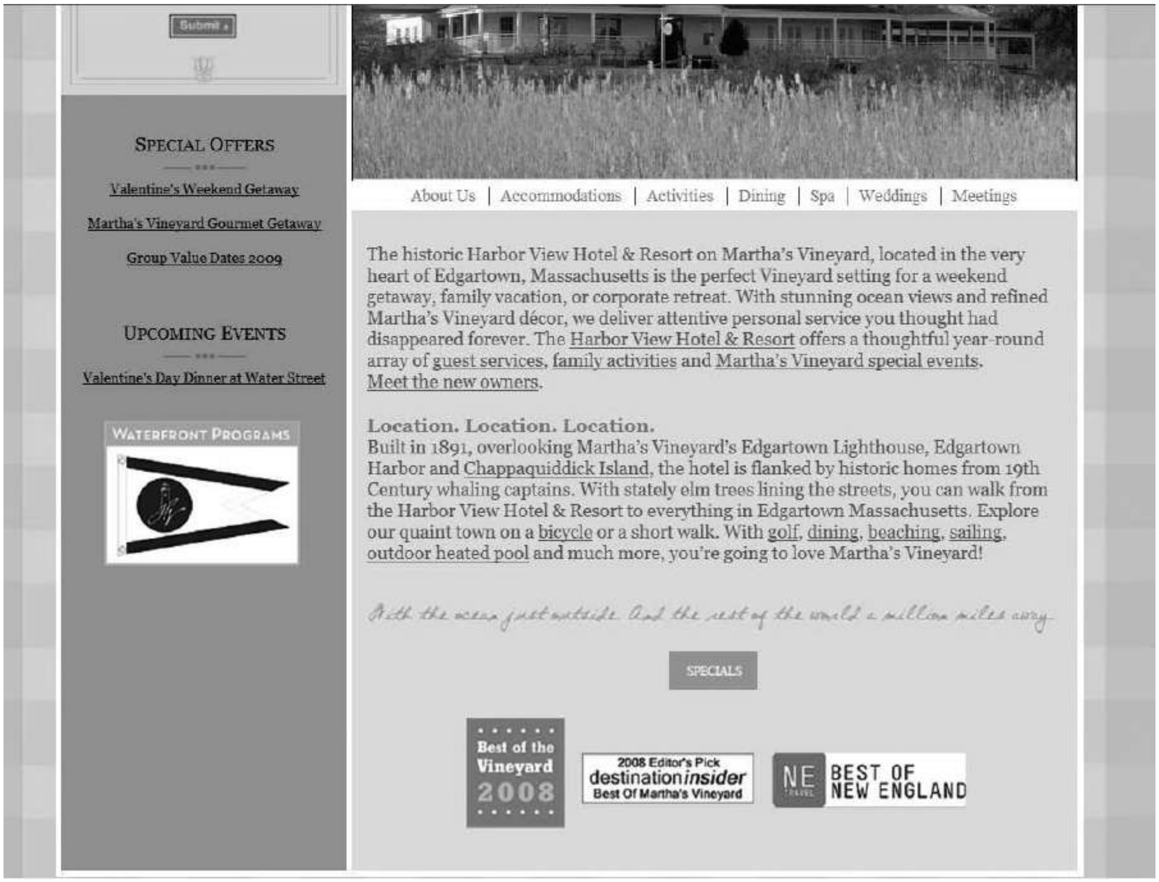
However, how would a visitor know this? There are no instructions and no visible control to click on. This is an example



The historic Harbor View Hotel & Resort on Martha's Vineyard, located in the very heart of Edgartown, Massachusetts is the perfect Vineyard setting for a weekend getaway, family vacation, or corporate retreat. With stunning ocean views and refined Martha's Vineyard décor, we deliver attentive personal service you thought had disappeared forever. The Harbor View Hotel & Resort offers a thoughtful year-round array of guest services, family activities and Martha's Vineyard special events. Meet the new owners.

Location. Location. Location.
Built in 1891, overlooking Martha's Vineyard's Edgartown Lighthouse, Edgartown

FIGURE 51.1 **(See color insert.)** Usability: invisible functionality. http://www.harbor-view.com/.


Submit »
SPECIAL OFFERS
Valentine's Weekend Getaway
Martha's Vineyard Gourmet Getaway
Group Value Dates 2009
UPCOMING EVENTS
Valentine's Day Dinner at Water Street
WATERFRONT PROGRAMS
About Us | Accommodations | Activities | Dining | Spa | Weddings | Meetings
SPECIALS
Best of the Vineyard 2008
2008 Editor's Pick destinationinsider Best Of Martha's Vineyard
NE BEST OF NEW ENGLAND

The historic Harbor View Hotel & Resort on Martha's Vineyard, located in the very heart of Edgartown, Massachusetts is the perfect Vineyard setting for a weekend getaway, family vacation, or corporate retreat. With stunning ocean views and refined Martha's Vineyard décor, we deliver attentive personal service you thought had disappeared forever. The Harbor View Hotel & Resort offers a thoughtful year-round array of guest services, family activities and Martha's Vineyard special events. Meet the new owners.

Location. Location. Location.
Built in 1891, overlooking Martha's Vineyard's Edgartown Lighthouse, Edgartown Harbor and Chappaquiddick Island, the hotel is flanked by historic homes from 19th Century whaling captains. With stately elm trees lining the streets, you can walk from the Harbor View Hotel & Resort to everything in Edgartown Massachusetts. Explore our quaint town on a bicycle or a short walk. With golf, dining, beaching, sailing, outdoor heated pool and much more, you're going to love Martha's Vineyard!

With the ocean just outside. And the rest of the world a million miles away.

FIGURE 51.2 Usability: affordances. http://www.harbor-view.com/.

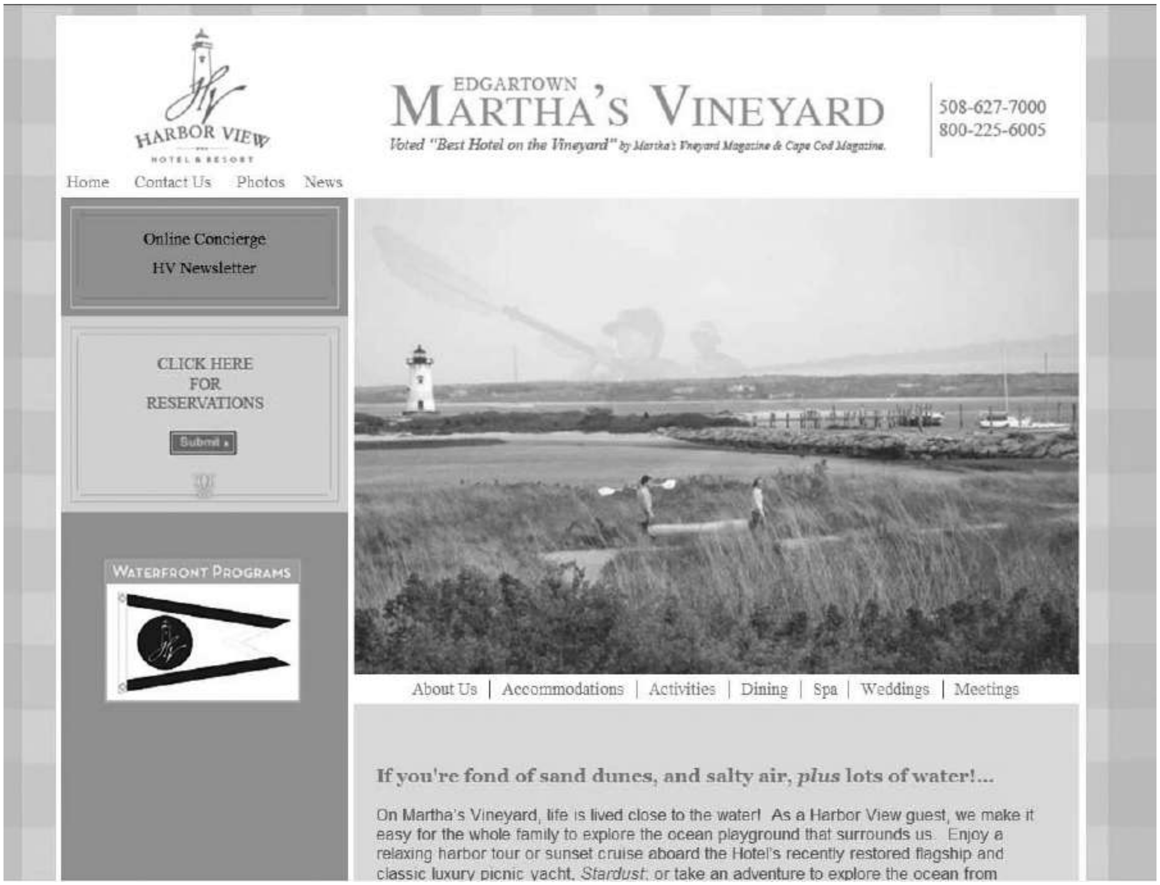
of a usability issue known as "invisible functionality"—there is nothing visible on the page to let visitors know that this functionality is available or how to invoke it. While the graphic design is appealing and the photos certainly contribute to the persuasiveness of the site, this invisible functionality represents a shortcoming in usability.

If visitors scroll down, as shown in Figure 51.2, they may note a number of images at the very bottom and in the left hand nav (navigation) bar. Are these just pictures or are they in fact active links? It is true that if visitors roll the cursor over an image, the cursor will let them know if an image is a link by changing shape (e.g., in MS IE, from an arrow to a hand icon). But there is nothing in the design of the images themselves to help the visitor distinguish between images that are links and images that are not. In fact, in this case, the "Best of the Vineyard" and "Best of New England" images are just images, while the other three are active links.

In addition, two of the image links shown in Figure 51.2 take you to another internal page on this site, while the third takes you outside this site to another website. Can you tell which does what? In fact, there is no cue to distinguish between internal and external links. In this case, the "Editor's Pick" image takes you to an external site, while the others take you to internal pages. In a related example, the logo shown at the top of Figure 51.1 is not a link. Since having a logo represent a link to a site's Home page has become a de facto standard on websites, this may violate visitors' expectations. Being able to quickly determine what on a web page is an active link and what is not, as well as which links are internal and which external, is an aspect of usability. Links designed to make it clear they are links and clarify important differences in their behaviors are said to have good "affordances."

Other examples of poor affordances can be seen back in Figure 51.1. In the left hand nav bar, sometimes all caps are used to designate headers (Special Offers, Upcoming Events), while other times they are used for text links (Click Here for Reservations). Also, sometimes text links are displayed with no underline but take on an underline when a visitor points to them (Online Concierge, HV Newsletter), while other text links are displayed underlined and do not change in any way when pointed to (Valentine's Weekend Getaway, Martha's Vineyard Gourmet Getaway). Generally speaking, there is very little consistency in the way text links are designed across this site, making it hard for visitors to learn and remember which text phrases are links and which are not.

Now imagine visitors to this hotel site go to an internal page, say the "Water Activities" page, which is available from a drop down menu from the "Activities" link on the main nav


HARBOR VIEW
HOTEL & RESORT
Home
Contact Us
Photos
News
EDGARTOWN
MARTHA'S VINEYARD
Voted "Best Hotel on the Vineyard" by Martha's Vineyard Magazine & Cape Cod Magazine.
508-627-7000
800-225-6005
Online Concierge
HV Newsletter
CLICK HERE FOR RESERVATIONS
Submit
WATERFRONT PROGRAMS
About Us | Accommodations | Activities | Dining | Spa | Weddings | Meetings

If you're fond of sand dunes, and salty air, *plus* lots of water!...

On Martha's Vineyard, life is lived close to the water! As a Harbor View guest, we make it easy for the whole family to explore the ocean playground that surrounds us. Enjoy a relaxing harbor tour or sunset cruise aboard the Hotel's recently restored flagship and classic luxury picnic yacht, *Stardust*; or take an adventure to explore the ocean from

FIGURE 51.3 Usability: "you are here" cue. http://www.harbor-view.com/activities_water.asp.

bar under the slide show area, shown in Figure 51.3. Let us say they scroll through this page reading for a while, then perhaps are interrupted for a while, then eventually return to viewing this page, scrolling back up to the top where they can see the main nav bar.

What page are they on? How did they get there? There is no way to tell. No meaningful page title. No breadcrumbs. Not even a cue in the main nav bar to tell them they are somewhere down the "Activities" link pathway. Cues that help visitors get oriented and learn their way around a site's information architecture are known as "you are here" cues. They are especially important for first time and infrequent visitors. This site lacks them. Visitors get lost, cannot find their way back to content of interest, and miss finding a lot of content on sites lacking "you are here" cues.

The above examples represent poor usability. Now let us visit a site that follows a number of principles of usability: the multiple-award-winning (http://www.crunchbase.com/company/netflix) mail order movie rental site Netflix (http://www.netflix.com/). Figure 51.4 shows the Netflix home page as of this writing.

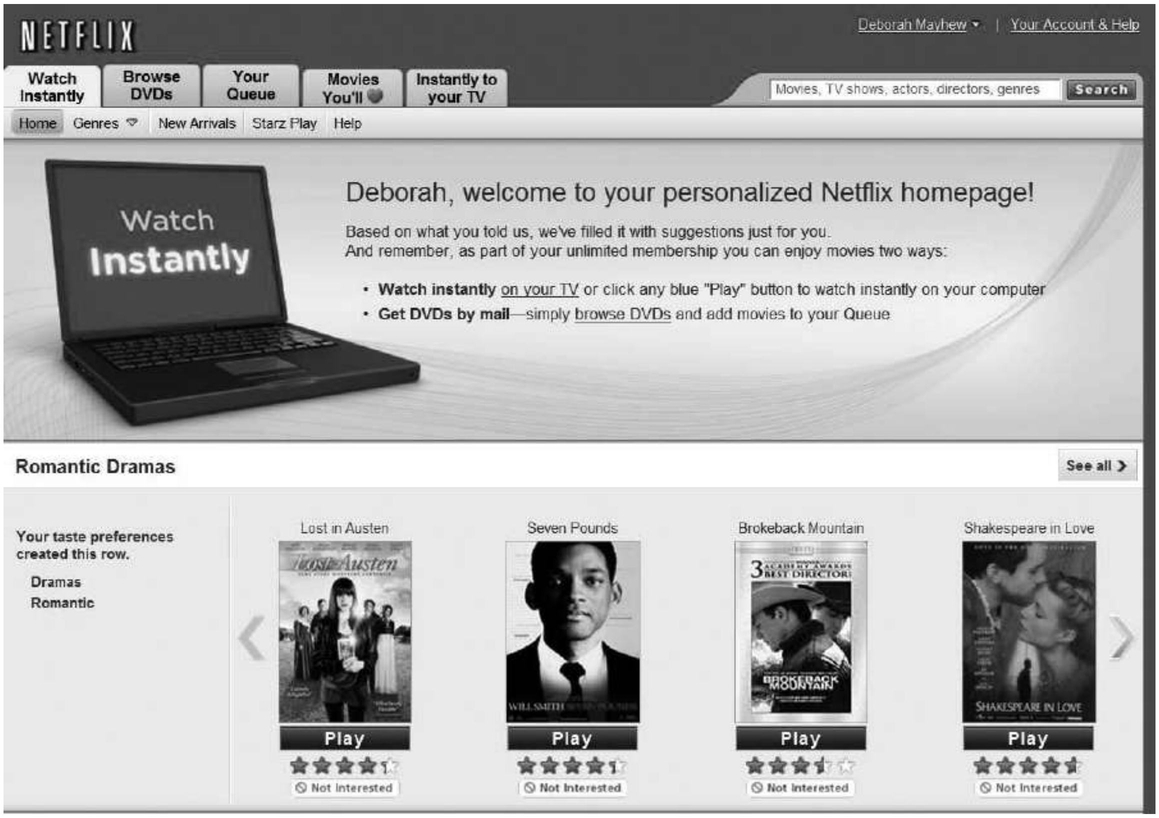
First, note the two levels of navigational links at the top and the embedded "you are here" cues that let you know where in the information architecture you are at the moment (white in the tabbed top level, gray in the second level menu bar). Next, note the personalization on this page once you are logged in. This is very helpful information to regular visitors, making it easier to find movies of interest in a huge product space. Now let us look in the visitor's movie queue, shown in Figure 51.5.

The pop up movie summary shown is invoked by hovering the cursor over a movie title in the list for about a second. No click is required to either invoke or close this pop up, just mouse movement. This provides a very efficient way to browse a little more content about movies, compared to using the movie title links to navigate to another page and then navigate back. Visitors can slowly drag their cursor down the list and get summaries of any movie quickly, then move on quickly, without losing the context of the whole list.

It is also true that if the visitor drags the cursor more quickly down the list, the pop ups *would not* come up, which is equally important, to avoid a lot of pop ups coming and going when the visitor is really just trying to move the cursor to another movie title.

51.2.2 Persuasiveness

Next let us look at some examples of eCommerce website persuasiveness. Recall that one important aspect of persuasiveness is establishing trust, especially for vendors that are



Deborah, welcome to your personalized Netflix homepage!

Based on what you told us, we've filled it with suggestions just for you.
And remember, as part of your unlimited membership you can enjoy movies two ways:

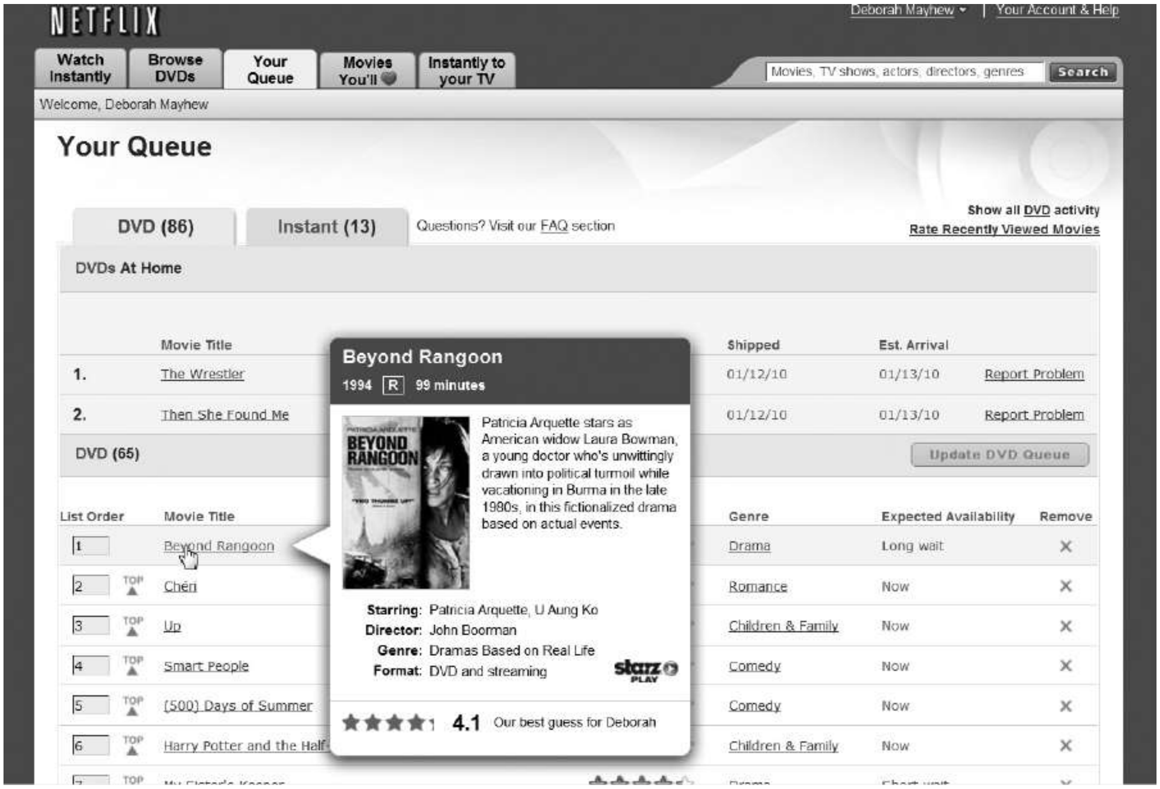
- **Watch instantly** on your TV or click any blue "Play" button to watch instantly on your computer
- **Get DVDs by mail**—simply browse DVDs and add movies to your Queue

Romantic Dramas

Your taste preferences created this row.
Dramas
Romantic

Lost in Austen | Seven Pounds | Brokeback Mountain | Shakespeare in Love

FIGURE 51.4 Usability: "you are here" cue. http://www.netflix.com/.



Welcome, Deborah Mayhew

Your Queue

Questions? Visit our FAQ section

Show all DVD activity
Rate Recently Viewed Movies

DVDs At Home

	Movie Title	Shipped	Est. Arrival	
1.	The Wrestler	01/12/10	01/13/10	Report Problem
2.	Then She Found Me	01/12/10	01/13/10	Report Problem

DVD (65)

List Order	Movie Title	Genre	Expected Availability	Remove
1	Beyond Rangoon	Drama	Long wait	✕
2	Chéri	Romance	Now	✕
3	Up	Children & Family	Now	✕
4	Smart People	Comedy	Now	✕
5	(500) Days of Summer	Comedy	Now	✕
6	Harry Potter and the Half	Children & Family	Now	✕

Beyond Rangoon
1994 R 99 minutes

Patricia Arquette stars as American widow Laura Bowman, a young doctor who's unwittingly drawn into political turmoil while vacationing in Burma in the late 1980s, in this fictionalized drama based on actual events.

Starring: Patricia Arquette, U Aung Ko
Director: John Boorman
Genre: Dramas Based on Real Life
Format: DVD and streaming

4.1 Our best guess for Deborah

FIGURE 51.5 **(See color insert.)** Usability: efficiency. http://www.netflix.com/Queue?lnkce=sntQu&lnkctr=mhbque.

not already well-branded and known to visitors. Figures 51.6 and 51.7 show the home page of a garden supply vendor (https://www.tomatogiant.com/).

Note that when you arrive on this site, while there is a reference to what looks like a company name (Gardener's Choice), there is no "About Us" page, no "Contact Us" page, and no corporate address. It is hard to judge who this vendor is, and in fact the website seems to offer only a single product. With website scams becoming more and more common, more and more visitors will be skeptical of unfamiliar brands and will be looking for evidence that a website vendor is trustworthy. This site does not do a good job of establishing trust.

Second, as visitors read through the product information on this page, they may find themselves a little confused about what this product actually is. Is it seeds? A partially grown plant? If a plant, how big: 1 inch? 1 foot? Do you grow it in what it comes in, in a pot, or in a garden? Indoors or outdoors? Most of the images on the page show individual tomatoes. Only when the visitor scrolls down to the bottom of the page do they see one small image of a tomato plant (and you cannot tell if it is growing in an included container, a pot, or a garden) and read in the fine print that the product comes in "nursery packs." The product is frequently referred to as a "tomato" or a "tree tomato," rather than as a "tomato tree" or "tomato plant," adding to the uncertainty. There is also a statement in the middle of the page that reads "Each Set You Order Contains Plants!" Each set of what? Tomatoes? Plants? All in all, it is likely a visitor will be left quite uncertain about what this product actually is. At best, it takes reading every word on the page to come to a conclusion about what it *probably* is. And if a visitor cannot tell who the vendor is and cannot tell what the product is, what is the likelihood of a sale?

By contrast, let us take a look at a site offering flower delivery. Figures 51.8 and 51.9 show the home page of the Pro Flowers website (http://www.proflowers.com/), which has a conversion rate of over 30% (http://www.grokdot-com.com/2009/03/18/top-10-online-retailers-by-conversion-rate-february-2009/).

Here you see both "About Us" and "Contact Us" links (although at the bottom of the page and perhaps not as noticeable as they should be), and if you follow them, you can read trust-building information, such as the fact that the company has been in business since 1998 and currently has over seven million customers (although apparently no physical address). On the home page there is a reference to an endorsement by the Wall Street Journal, a very credible entity, and also a prominently displayed phone number at the top. These elements help establish credibility and trust.

In addition, if visitors drill down to inspect particular flower arrangements, there is ample product information including excellent photographs of the arrangements and vase choices, and listings of exactly how many of what types of flowers and greens are included. The combination of evidence of trustworthiness and adequate product information with attractive and clear product photographs surely accounts in part for the very high conversion rate on this website.

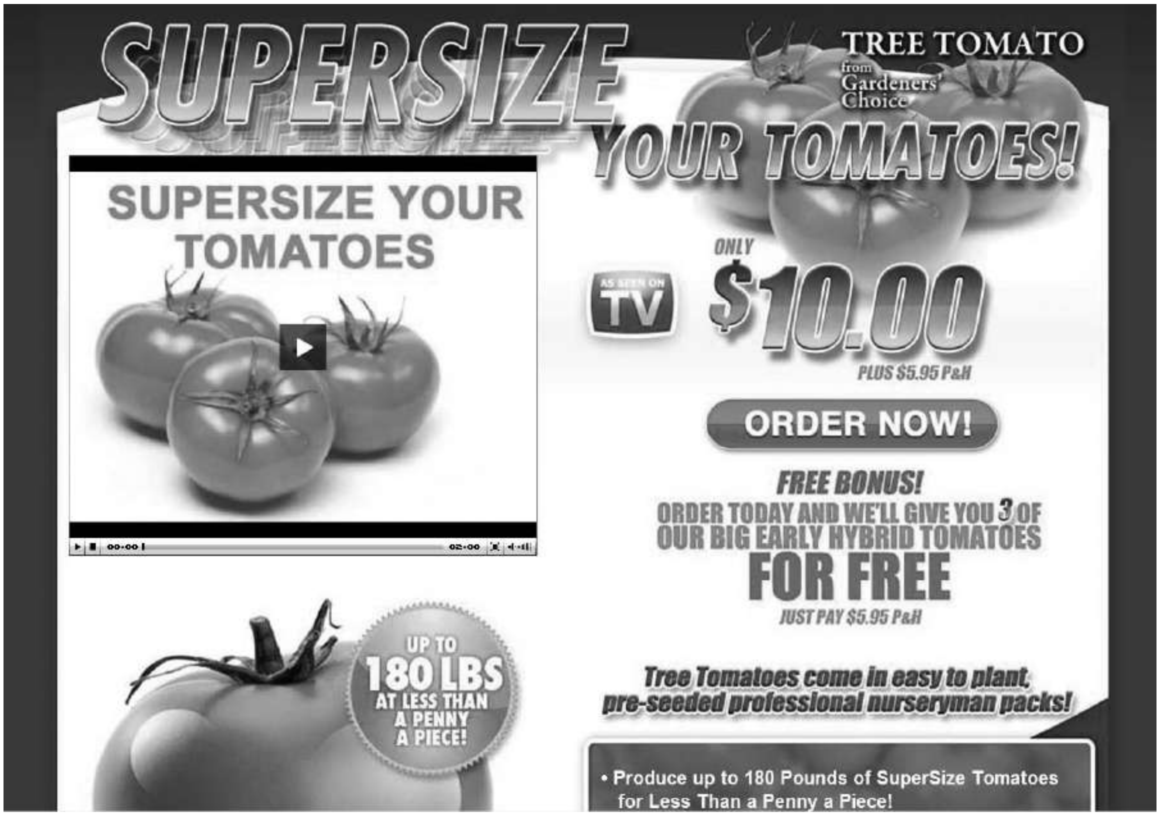
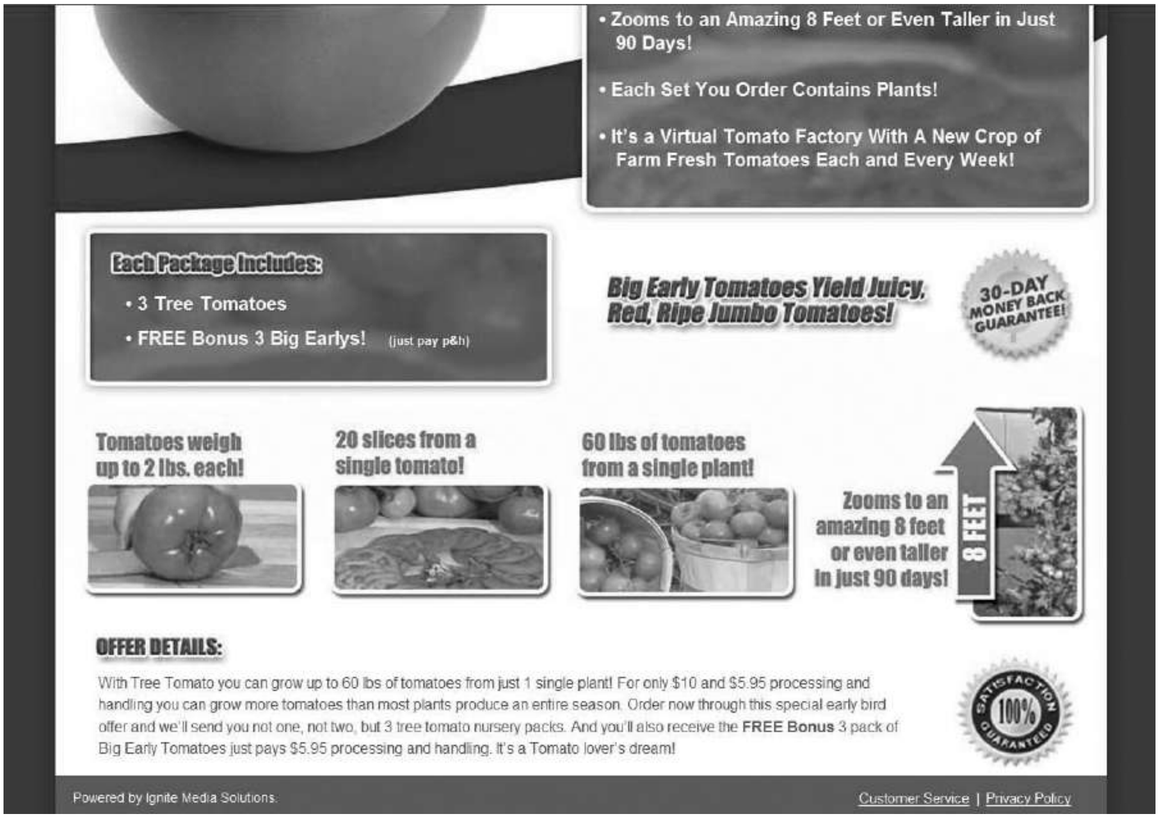

SUPERSIZE
TREE TOMATO
from Gardeners' Choice
YOUR TOMATOES!
SUPERSIZE YOUR TOMATOES
00:00
00:00
AS SEEN ON TV
ONLY
$10.00
PLUS $5.95 P&H
ORDER NOW!
FREE BONUS!
ORDER TODAY AND WE'LL GIVE YOU 3 OF OUR BIG EARLY HYBRID TOMATOES
FOR FREE
JUST PAY $5.95 P&H
UP TO 180 LBS AT LESS THAN A PENNY A PIECE!
Tree Tomatoes come in easy to plant, pre-seeded professional nurseryman packs!
• Produce up to 180 Pounds of SuperSize Tomatoes for Less Than a Penny a Piece!

FIGURE 51.6 Persuasiveness: trust. https://www.tomatogiant.com/.



- Zooms to an Amazing 8 Feet or Even Taller in Just 90 Days!
- Each Set You Order Contains Plants!
- It's a Virtual Tomato Factory With A New Crop of Farm Fresh Tomatoes Each and Every Week!

Each Package Includes:

- 3 Tree Tomatoes
- FREE Bonus 3 Big Earlys! (just pay p&h)

Big Early Tomatoes Yield Juicy, Red, Ripe Jumbo Tomatoes!

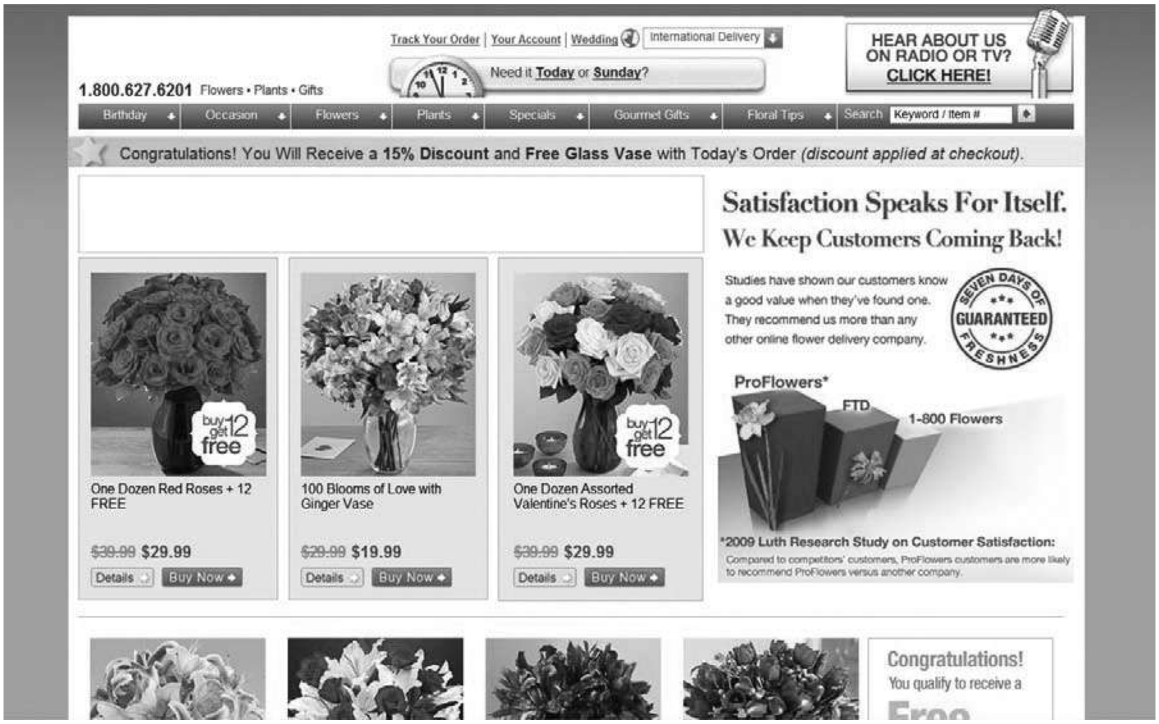
30-DAY MONEY BACK GUARANTEE!
Tomatoes weigh up to 2 lbs. each!
20 slices from a single tomato!
60 lbs of tomatoes from a single plant!
Zooms to an amazing 8 feet or even taller in just 90 days!
8 FEET
SATISFACTION 100% GUARANTEED

OFFER DETAILS:

With Tree Tomato you can grow up to 60 lbs of tomatoes from just 1 single plant! For only $10 and $5.95 processing and handling you can grow more tomatoes than most plants produce an entire season. Order now through this special early bird offer and we'll send you not one, not two, but 3 tree tomato nursery packs. And you'll also receive the **FREE Bonus** 3 pack of Big Early Tomatoes just pays $5.95 processing and handling. It's a Tomato lover's dream!

Powered by Ignite Media Solutions. Customer Service | Privacy Policy

FIGURE 51.7 Persuasiveness: trust. https://www.tomatogiant.com/.



Track Your Order | Your Account | Wedding | International Delivery
1.800.627.6201 Flowers • Plants • Gifts
Need it Today or Sunday?
HEAR ABOUT US ON RADIO OR TV? CLICK HERE!
Birthday
Occasion
Flowers
Plants
Specials
Gourmet Gifts
Floral Tips
Search Keyword / Item #
buy 12 get 12 free
Details
Buy Now
SEVEN DAYS OF GUARANTEED FRESHNESS
ProFlowers*
FTD
1-800 Flowers

Congratulations! You Will Receive a **15% Discount** and **Free Glass Vase** with Today's Order *(discount applied at checkout)*.

One Dozen Red Roses + 12 FREE ~~$39.99~~ $29.99

100 Blooms of Love with Ginger Vase ~~$29.99~~ $19.99

One Dozen Assorted Valentine's Roses + 12 FREE ~~$39.99~~ $29.99

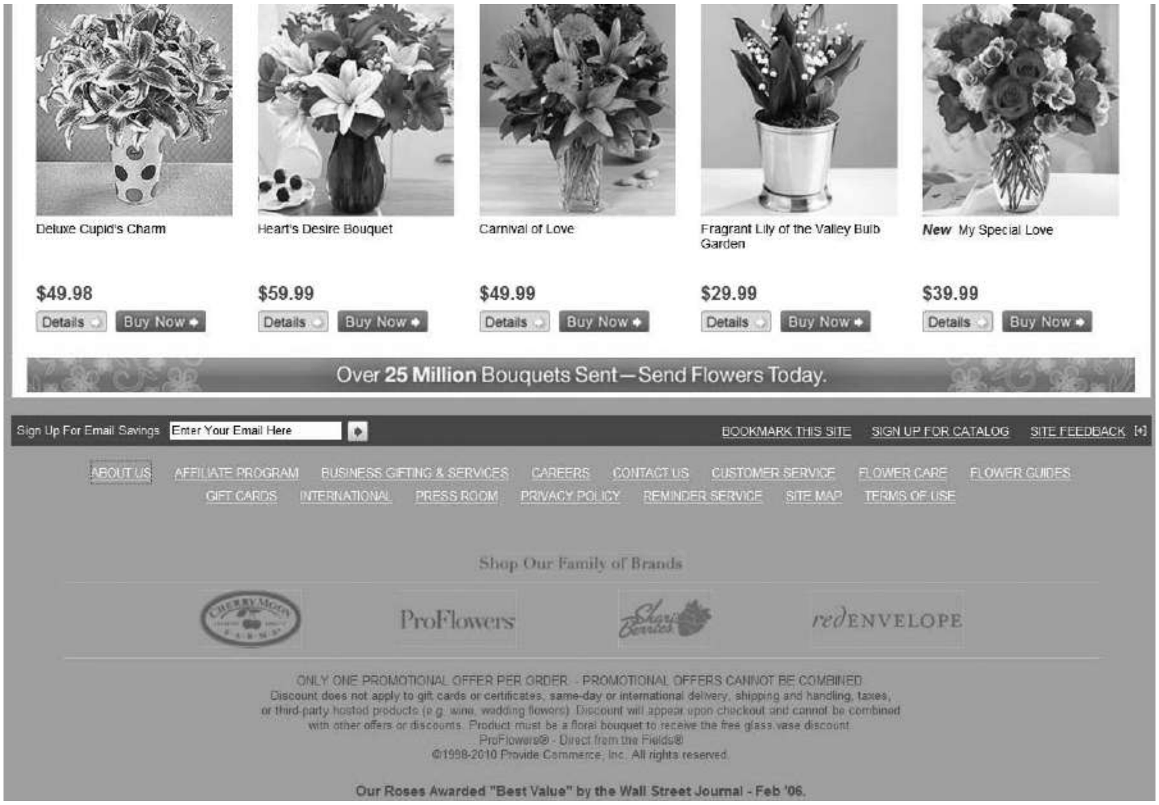
Satisfaction Speaks For Itself. We Keep Customers Coming Back!

Studies have shown our customers know a good value when they've found one. They recommend us more than any other online flower delivery company.

*2009 Luth Research Study on Customer Satisfaction: Compared to competitors' customers, ProFlowers customers are more likely to recommend ProFlowers versus another company.

Congratulations! You qualify to receive a Free

FIGURE 51.8 Persuasiveness: product information. http://www.proflowers.com/.



Deluxe Cupid's Charm	Heart's Desire Bouquet	Carnival of Love	Fragrant Lily of the Valley Bulb Garden	*New* My Special Love
$49.98	$59.99	$49.99	$29.99	$39.99

Over **25 Million** Bouquets Sent—Send Flowers Today.

ONLY ONE PROMOTIONAL OFFER PER ORDER - PROMOTIONAL OFFERS CANNOT BE COMBINED
Discount does not apply to gift cards or certificates, same-day or international delivery, shipping and handling, taxes, or third-party hosted products (e.g. wine, wedding flowers). Discount will appear upon checkout and cannot be combined with other offers or discounts. Product must be a floral bouquet to receive the free glass vase discount.
ProFlowers® - Direct from the Fields®
©1998-2010 Provide Commerce, Inc. All rights reserved.

Our Roses Awarded "Best Value" by the Wall Street Journal - Feb '06.

FIGURE 51.9 Persuasiveness: product information. http://www.proflowers.com/.

51.2.3 Graphic Design

Finally, let us look at some examples of graphic design, starting with the website of a well-known and very successful vendor of gourmet coffee: Starbuck's (http://www.starbucks.com/). Figure 51.10 shows the Starbuck's home page.

The graphic design on this page seems unlikely to invoke any positive emotions or associations. It is boring, pedestrian, and uninspired. It fails to exploit the possibility of creating a positive, appealing coffee shop "ambiance"—the dominant imagery does not even include coffee or a coffee shop. Instead, it is very businesslike and likely leaves visitors cold. Internal pages on the site do no better, the design is very perfunctory.

By contrast, consider the home page of a competitor's website: New England Coffee (http://www.newenglandcoffee.com/), shown in Figures 51.11 and 51.12.

The imagery on this home page immediately conjures up the delights of good coffee in a way that is even season-specific (the screen shot was taken in mid-winter). The photographs create images in the mind and evoke smells, tastes, and even the pleasurable sensation of heat. The brown and cream color palette—more apparent on internal pages such as that shown in Figure 51.12—help to evoke the experience of good coffee. Reputation, usability, and persuasion aside, the New England Coffee site is simply much more enjoyable to visit than the Starbuck's site and more effective at invoking an appetite for good coffee.

51.3 MEASURING THE IMPACT OF THE USER EXPERIENCE

The impact of the quality of eCommerce website *usability* on the bottom line is well established. *Cost Justifying Usability—An Update for the Internet Age* (Bias and Mayhew 2005) provides many examples. Several impressive anecdotes are reported in one chapter (Rohn 2005). According to the first, Dell Computer applied usability principles to an eCommerce site, which resulted in sales increasing from $1 million to $34 million per day within 6 months. In a second, Skechers (a shoe vendor) moved its product selection closer to the home page, resulting in a sales increase of over 400%. And in a third, IBM invested in a site redesign, resulting in a 400% sales increase and an 84% decrease in use of the Help button.

There is also ample research to support the importance of *persuasion* and *graphic design* principles as well, although not much of it was conducted specifically in the context of eCommerce website design. While we do not have research that teases out the impact of each of these three qualities on eCommerce conversion rates, it seems likely that they are somewhat cumulative, that is, that adding in optimized

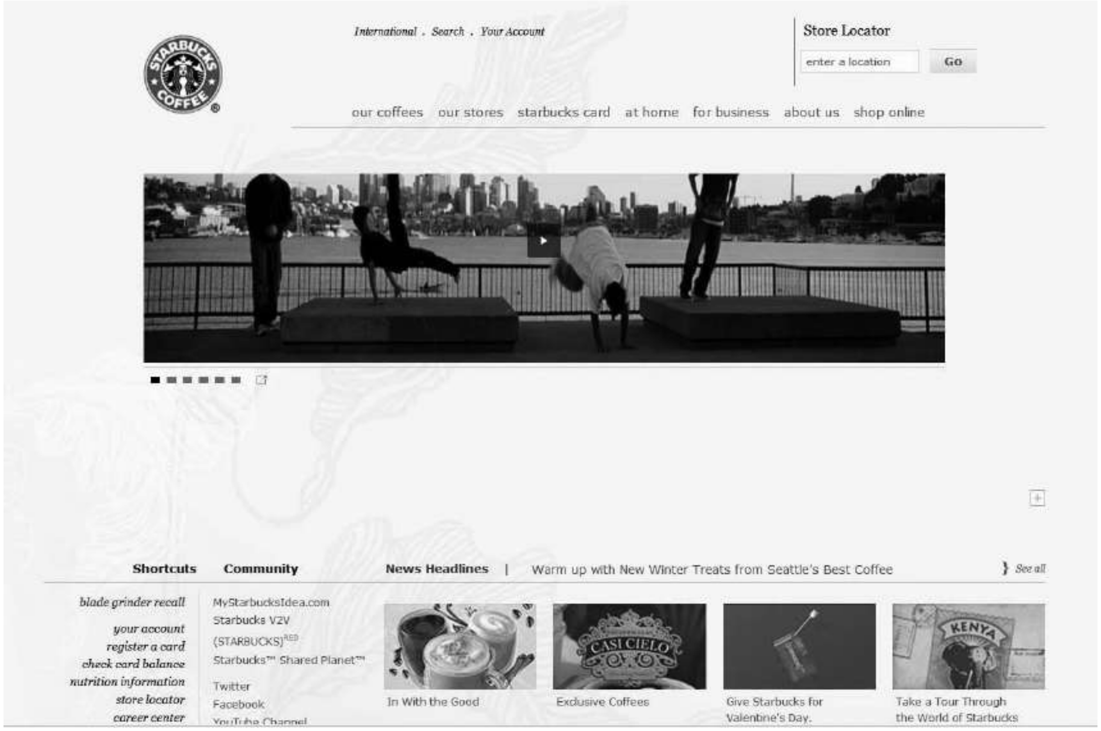
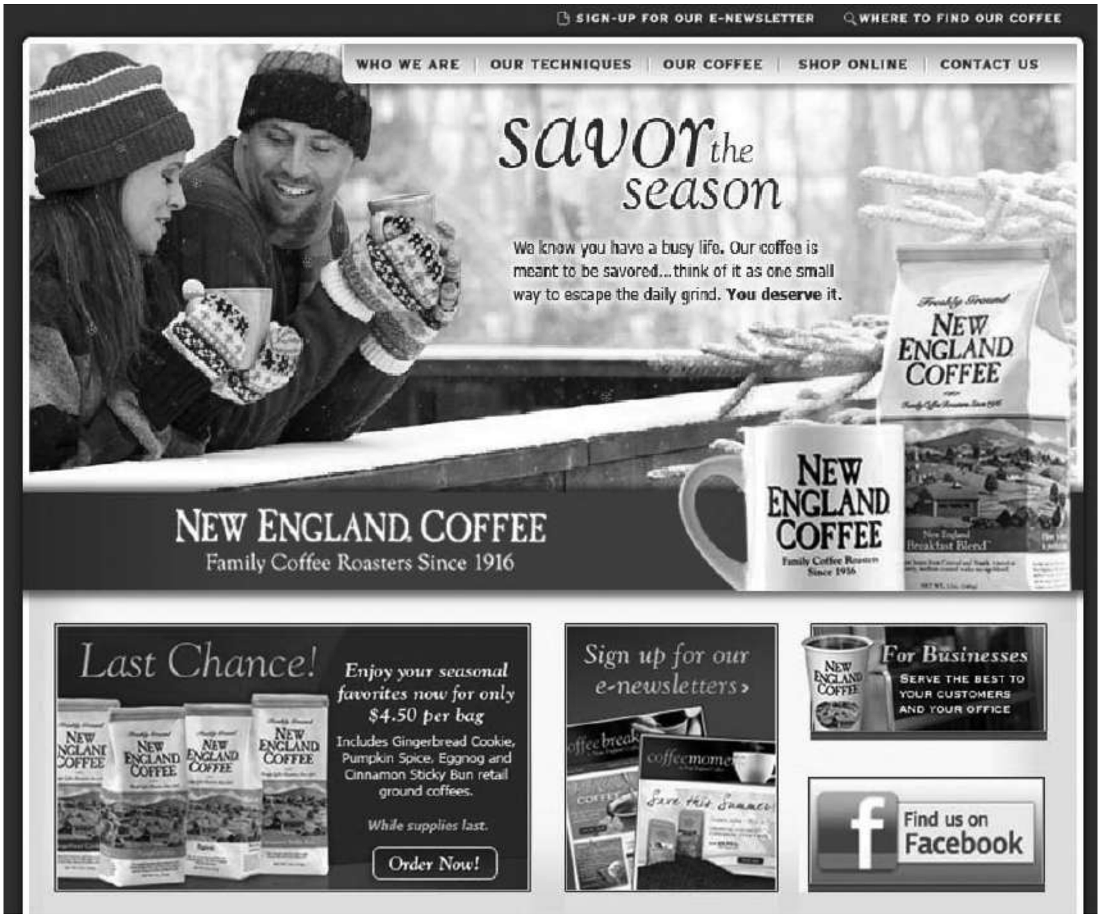


FIGURE 51.10 Graphic design: ambiance. http://www.starbucks.com/.



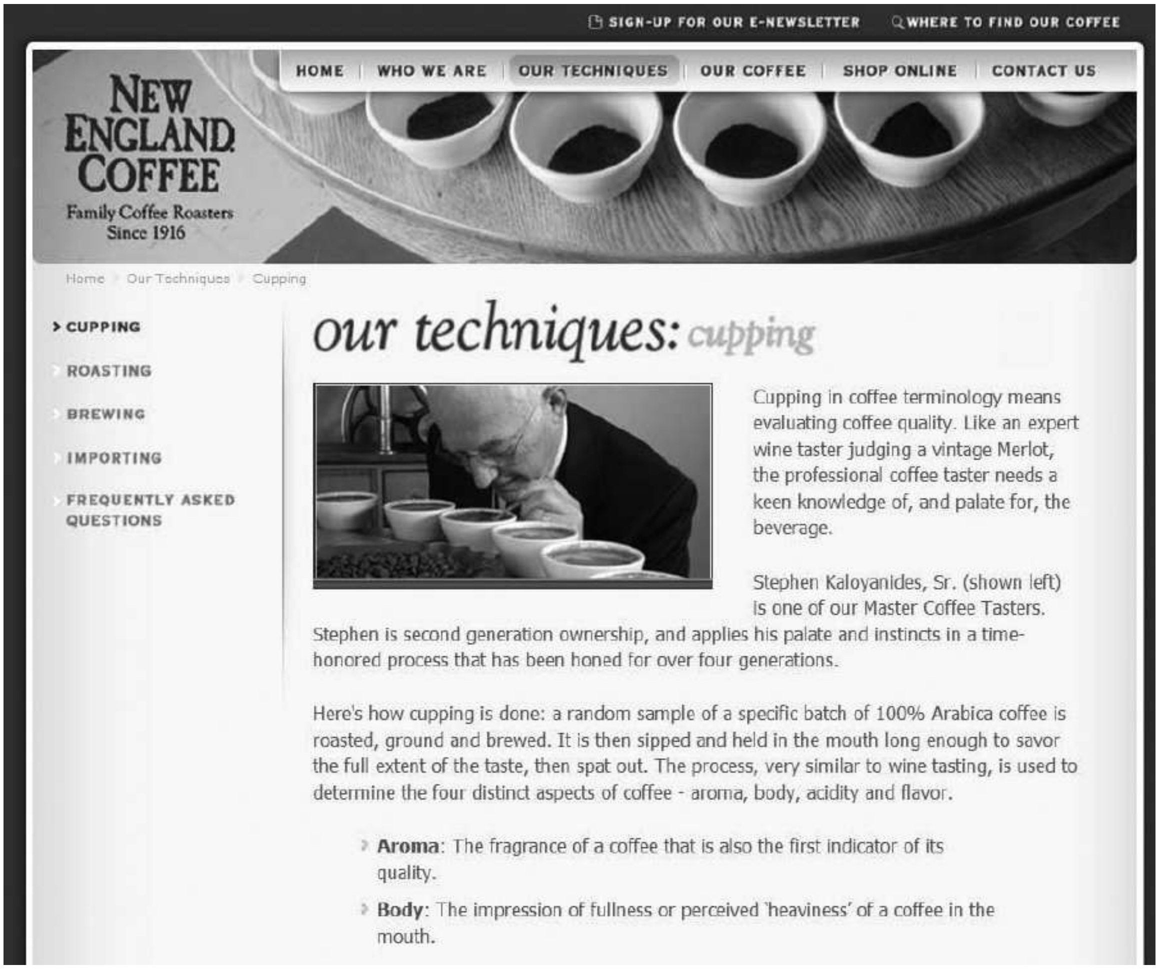
We know you have a busy life. Our coffee is meant to be savored...think of it as one small way to escape the daily grind. **You deserve it.**

Enjoy your seasonal favorites now for only $4.50 per bag

Includes Gingerbread Cookie, Pumpkin Spice, Eggnog and Cinnamon Sticky Bun retail ground coffees.

While supplies last.

FIGURE 51.11 Graphic design: ambiance. http://www.newenglandcoffee.com/.



our techniques: cupping

Cupping in coffee terminology means evaluating coffee quality. Like an expert wine taster judging a vintage Merlot, the professional coffee taster needs a keen knowledge of, and palate for, the beverage.

Stephen Kaloyanides, Sr. (shown left) is one of our Master Coffee Tasters. Stephen is second generation ownership, and applies his palate and instincts in a time-honored process that has been honed for over four generations.

Here's how cupping is done: a random sample of a specific batch of 100% Arabica coffee is roasted, ground and brewed. It is then sipped and held in the mouth long enough to savor the full extent of the taste, then spat out. The process, very similar to wine tasting, is used to determine the four distinct aspects of coffee - aroma, body, acidity and flavor.

- **Aroma**: The fragrance of a coffee that is also the first indicator of its quality.
- **Body**: The impression of fullness or perceived 'heaviness' of a coffee in the mouth.

FIGURE 51.12 Graphic design: ambiance. http://www.newenglandcoffee.com/techniques/cupping/index.asp.

persuasion and graphic design could improve conversions even more than optimizing usability alone.

In an example completely justified by research and experience (see Bias and Mayhew 2005), suppose an eCommerce website for a small business currently has the following:

- 1500 monthly visitors (a very small business)
- A 2% conversion rate
- An average $50 in revenue per online order

Annual revenue from this site is thus on average 1,500 × 0.02 × $50 × 12, or $18,000.

A very modest prediction would be that improving the *usability* of the site could increase the conversion rate from 2% to 3%, which would result in an annual revenue of 1,500 × 0.03 × $50 × 12, or $27,000, an increase of $9,000 in revenues. It seems likely that optimizing *persuasiveness* and *graphic design* as well could bump this conversion rate up even higher, increasing conversions and revenues even more.

What we do know is that statistically speaking, there is a great deal of opportunity to improve conversion rates on eCommerce websites. According to http://index.fireclick.com, the average eCommerce conversion rate in March 2009 was 1.87%. That is, more than 98% of visitors left eCommerce websites without buying. Similarly, the same source cites the shopping cart abandonment rate as 40%, meaning that 40% of visitors left sites even after putting items in their shopping cart. While some of these conversion failures will be due to things like unqualified leads coming to the site (i.e., lack of *utility* to incoming traffic) or lack of *functional integrity*, surely some significant percent are due to suboptimal usability, persuasiveness, and/or graphic design.

51.4 DEFINING A PROCESS FOR GREAT eCOMMERCE USER EXPERIENCE DESIGN

Over 10 years of experience with website developers suggests that in many organizations, sites typically evolve the quality of their user experience over releases, in a particular order represented in the pyramid in Figure 51.13.

Often a business starts with a plan for products or services and a target market, which may or may not be supported by research or already established in a bricks and mortar business. Then they may focus on getting up a functioning eCommerce website as soon as possible. If the business already exists, often the website is created directly from brochures and other traditional marketing material and catalogs, with

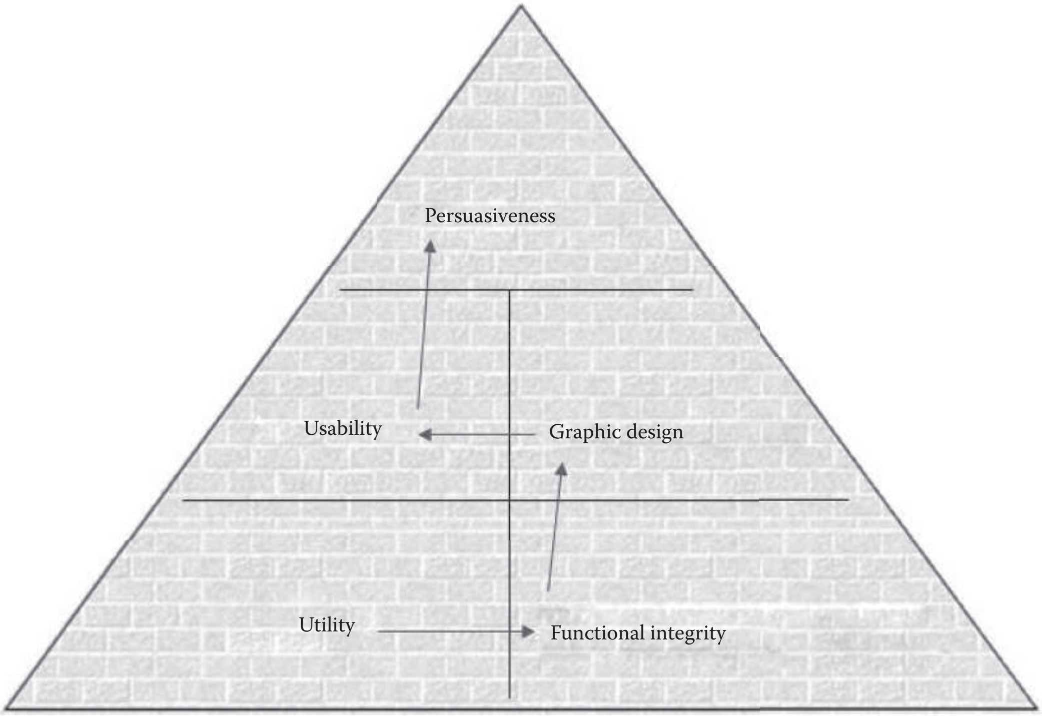


FIGURE 51.13 Typical website user experience design evolution.

little attention paid to the online user experience. That is, they start with the basics of *utility* and *functional integrity*, and launch.

Of the three remaining qualities of user experience, the one most familiar to web developers is *graphic design*, so often this is the first thing addressed in enhancements once a site is launched. Bricks and mortar businesses in particular are already familiar with the importance of good graphic design from their hardcopy marketing materials designers, and so this is a natural next step. It may take a business a while to figure out that what works on hardcopy does not necessarily work as a part of an online user experience, but at least as an organization they may be familiar with the value of graphic design and have the graphic design skill set already in house.

Less familiar to web developers and their organizations is the quality and discipline of *usability*. Usually this quality is not pursued until and unless there is some sort of "pain," such as an inordinate amount of website customer support calls or high bounce rates and low conversion rates.

Finally, in spite of the fact that established business organizations usually use marketing and sales professionals, *persuasiveness* is the least familiar user experience design quality to web developers, and even graphic design and usability experts may be unaware of this field and the importance of this quality to user experience design. Thus, this quality is often the last addressed or never addressed at all.

Evolving a website's user experience in the typical order described above and in Figure 51.13 is inefficient and expensive. It is more efficient and effective as well as less expensive to use a design process that addresses all the user experience qualities right from the start and in a logical order. The optimal order in which to address the five user experience qualities is illustrated in Figure 51.14. As the figure implies, designing and building a house provides a good metaphor for designing and building a website.

Utility does make sense as a first step. Every house building project starts with a "blueprint," which captures the basic needs and desires of the future occupants, such as the need for a certain number of bedrooms and the desire for a dining room large enough to accommodate a certain number of people.

Similarly, every business—traditional or web-based—should have a clear and well researched business plan that defines products or services, intended market, competition, competitive edge, and so on.

In homebuilding, an early next step involves designing a foundation, which will support the framework of the house, given the blueprint. No choices of building materials or interior decoration are necessary at this point. Similarly in web design and development, *usability* requirements are incorporated into an information architecture "wireframe," which is the foundation of a user experience design. No persuasive elements or graphic design are specified at this point. Without basic usability, they will not be very effective.

Next, the outside of the house is designed—shingles or brick, roofing color and material, doors and windows. These are all designed to make a home appealing from the outside, you might say to make it attract entry. Similarly in eCommerce website design, *persuasive* design elements are laid

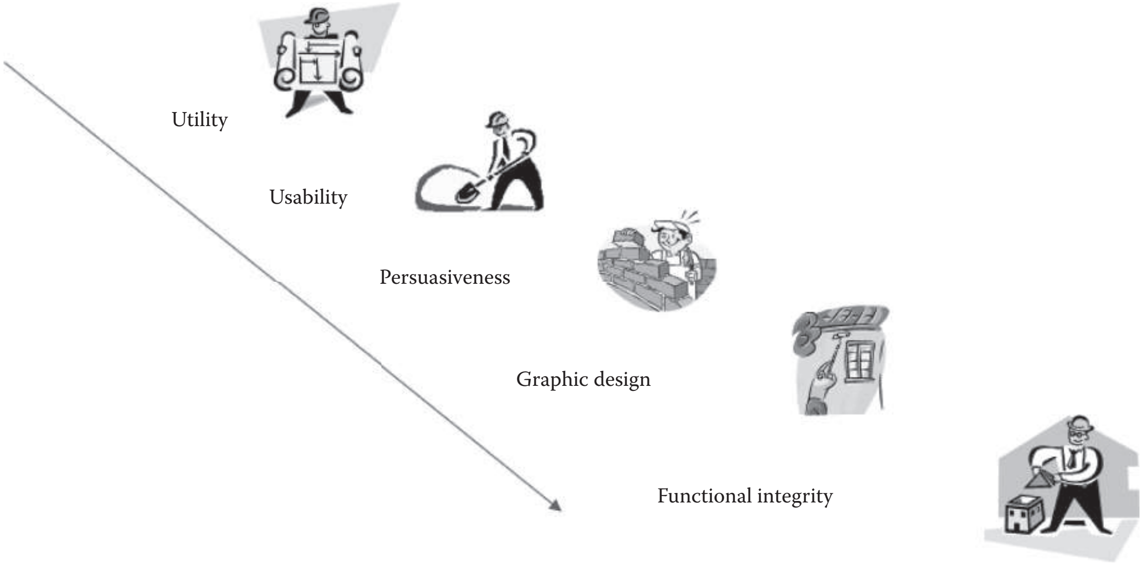


FIGURE 51.14 Recommended user experience design process.

over the information architecture design. Certain persuasive design elements appear on a site "home" page, where their purpose is to motivate engagement and encourage entry into the internal pages of the site. As illustrated earlier, persuasive elements that create trust and encourage engagement—such as credible recognitions and awards, company history, a physical business address, testimonials and the like—go beyond usability, but are more fundamental than the graphic design of those elements.

Once the framework, foundation, and exterior of a house are designed, decoration of the interior is planned to appeal to the tastes of the future occupants. Curtains, furniture, lighting, rugs, and the like move the house beyond adequate and usable, to homey, appealing, and an expression of personality and taste. Likewise, the *graphic design* of a website is the dressing over the framework of usability and persuasiveness. It makes the difference between a functional "house" and an emotionally satisfying "home."

Finally, the house—or website—is built. This is where *functional integrity* comes into the process. Just as the house is fully designed before it is built, it is premature and potentially disastrous to build a website before it has been thoroughly and well designed. You do not want to be well into decorating your bedrooms before discovering that you do not have enough for your family or build your dining room only to find that a table for 12 simply would not fit in it. Similarly, you do not want to be doing graphic design and building your eCommerce website before you are sure it will serve the needs of your visitors (usability) and engage them and encourage conversions (persuasiveness).

Getting the detailed design right before implementation is invariably more cost-effective than redesigning after launch. Adding a bedroom or enlarging the dining room is a lot cheaper and faster on paper than it is after the house is already built. In the case of eCommerce websites, if stakeholders rush to launch with little investment in the user experience and then redesign and relaunch later with an improved user experience, not only will it be more expensive, but in addition, visitors who had a bad experience with your first launch may simply never return.

51.5 SUMMARY

To summarize, while the term "user interface" is heavily associated with the specific quality of usability, the term "user experience" is more useful to eCommerce web designers and developers because it encompasses all the elements that impact a visitor's experience using a website, and in particular that impact the likelihood of a conversion. This chapter offered an optimal process (at a high level) for designing great user experiences and identified the set of skills that will increase the likelihood of success. In addition, it offered the insight that an eCommerce website development project team needs not only to be an interdisciplinary team, but also to be a team of members who all know how to work collaboratively with very different disciplines to achieve the common goal of a great user experience.

REFERENCES

Bias, R., and D. J. Mayhew. 2005. *Cost Justifying Usability—An Update for the Internet Age*. San Francisco, CA: Morgan Kaufmann Publishers, Inc.

Rohn, J. 2005. Cost justifying usability in vendor companies. In *Cost Justifying Usability—An Update for the Internet Age*, ed. R. Bias and D. J. Mayhew, 185–214. San Francisco, CA: Morgan Kaufmann Publishers, Inc.

52 Human–Computer Interaction and Software Engineering for User Interface Plasticity

Joëlle Coutaz and Gaëlle Calvary

CONTENTS

52.1 INTRODUCTION

Human–computer interaction (HCI) and software engineering (SE) are like two old friends with different backgrounds: they share values but use them differently. Both domains address the design and development of useful and usable systems and are concerned with “requirements analysis,” “incremental and iterative design,” as well as “quality assurance.” However, they address these problems with different development processes, different notations, and different priorities. For HCI, human is the first-class entity in all phases of the development. For SE, the final objective is a running system developed at minimal cost and delivered in time, while satisfying contractual specifications. The user is involved, at best, at the very beginning of the process and hopefully at the very end of the project for summative evaluation. However, to avoid or correct wrong design decisions, this is too little and too late. Even in the early stages of development, functional requirements and quality goals are rarely the result of a close collaboration between HCI and SE specialists.

There are many reasons for the lack of collaboration between HCI and SE scientists and practitioners: mutual ignorance resulting from educational background, and from there, economic consideration. HCI methods such as contextual design, scenario-based approaches (Rosson and Carroll 2002), and task analysis are perceived as too demanding in terms of time and competence to inform system requirements in a formal and timely manner. On the other hand, Unified Modeling Language (UML) use cases, which express the functions that the system should support with a scenario-based flavor, are pale attempts to factor out user-centered concerns. They do not result from a human-centered requirements analysis nor do they have the expressive power of task models. Task-modeling techniques such as Concur Task Tree (Paternò 2003) or User Action Notation (Hartson, Siochi, and Hix 1990), which use notations familiar to computer scientists (i.e., Language of Temporal Ordering Specification operators and logic), are not used in SE. On the other hand, task models are not well suited for expressing what can go wrong, whereas the SE KAOS goal-oriented modeling approach supports the explicit expression of "goal obstacles" (van Lamsweerde 2009). Similarly, domain-dependent concepts referenced in task models are ill defined, whereas UML class diagrams would improve task specifications significantly.

In summary, HCI and SE pursue the same goal, using development processes and notations that sometimes overlap and complement each other. In this chapter, we present one way to exploit both fields for the development of plastic user interfaces (UIs) using the notion of model as the keystone between these two disciplines. This chapter is structured as follows: First, we define the concept of UI plasticity and develop the problem space for this concept. Exemplars of plastic interactive systems will illustrate aspects of this problem space. We then introduce the key objectives and principles of model-driven engineering (MDE) (http://planetmde.org) and analyze the contributions and limitations of MDE to address the problem of UI plasticity. Drawing from our experience with MDE applied to UI plasticity, we show how to address these limitations and conclude with recommendations for a research agenda.

52.2 USER INTERFACE PLASTICITY: DEFINITION

The term "plasticity" is inspired from the capacity of biological tissues such as plants and brain to undergo continuous deformation to rebuild themselves and to adapt to external constraints to preserve function without rupture. Applied to interactive systems, UI plasticity is the capacity of UIs to adapt to the context of use while preserving usability (Thevenin and Coutaz 1999) or human values (Calvary et al. 2003; Cockton 2004).* In the following sections, we define context of use, usability and the notion of human values in more detail.

52.2.1 Context and Context of Use

Since the early 1960s, the notion of context has been modeled and exploited in many areas of informatics. The scientific community has debated definitions and uses for many years without reaching a clear consensus (Dourish 2001; Dey 2001). Nonetheless, it is commonly agreed that context is about evolving, structured, and shared information spaces (Winograd 2001) and that such spaces are designed to serve a particular purpose (Coutaz et al. 2005). In UI plasticity, the purpose is to support the adaptation process of the UI to preserve use. Thus, there is no such thing as "the" context, but there is a context qualified by the process it serves. This is why we use the term "context of use" and not simply the word "context." A context change could be defined as the modification of the value of any element of the contextual information spaces. This definition would lead to an explosion of contexts. The following ontological foundation provides some structure to master this explosion.

52.2.1.1 Ontological Foundation for Context

As shown in Figure 52.1, a contextual information space is modeled as a directed graph where a node denotes a context and an edge denotes a condition to move between two contexts. In turn, a context is a directed graph of situations where a node denotes a situation and an edge denotes a condition to move between two situations. Thus, a contextual information space is a two-level data structure, that is, a graph of contexts where each context is in turn a graph of situations. If more structure is needed, situations may in turn be refined into "sub-situations," and so on. We now need to specify the domain of definition of contexts and situations.

A context is defined over a set E of *entities*, a set Ro of *roles* (i.e., functions) that these entities may satisfy, and a set Rel of *relations* between entities. Entities, roles, and relations are modeled as expressions of *observables* that are captured and/or inferred by the system. For example, in a conference room, E denotes the participants, Ro denotes the roles of a speaker and a listener, and Rel denotes some spatial relations between entities such as "entity e1 (who plays the role of a speaker) stands in front

* The cloud computing community uses the term "elasticity" for systems where "capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time" (NIST's definition—http://csrc.nist.gov/groups/SNS/cloud-computing/). Indeed, elasticity has the property of returning to an initial form (or state) following strain, which is not necessarily the case for plasticity. An elastic matter can break whereas a plastic entity aims at preserving survival. For these reasons, we consider that plasticity is a more general and demanding property than elasticity. We must admit however, that they are very similar in spirit. (In economics, elasticity measures the incidence of the variation of one variable on that of another variable. cf. Wikipedia.)

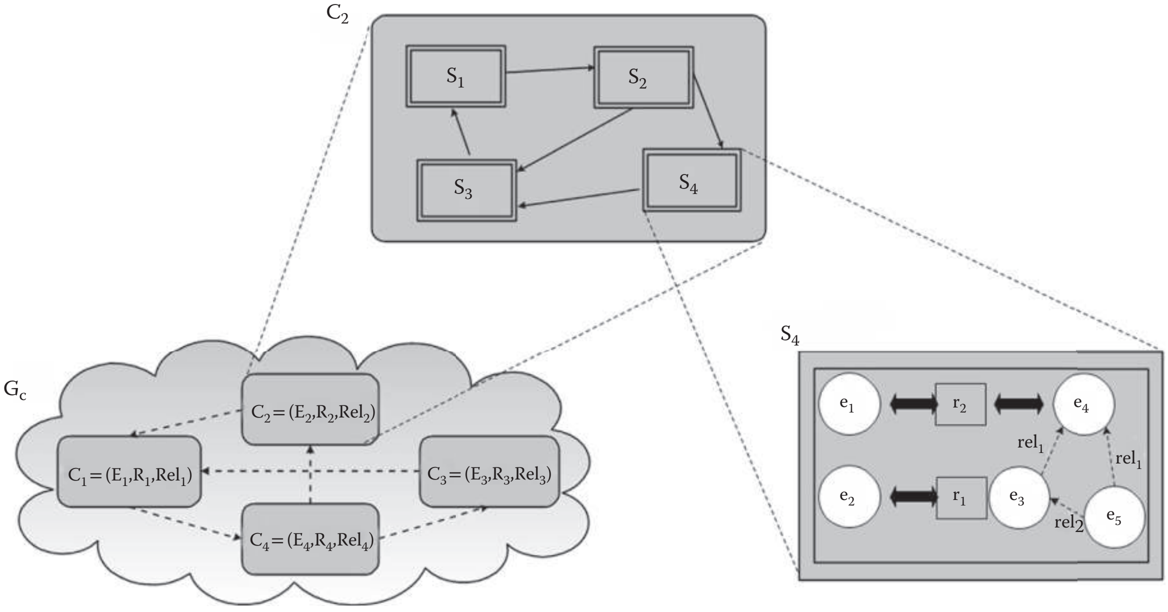


FIGURE 52.1 The graph of contexts G_C is composed of four contexts C_1, C_2, C_3, and C_4 defined on their own sets of entities, roles, and relations. In turn, Context C_2 is composed of four situations S_1, S_2, S_3, and S_4. By definition, these situations share the same sets of entities, roles, and relations. In S_4, entities e_1 and e_4 (elements of E_2) play the role r_2 (element of R_2), whereas role r_1 is played by entity e_2; e_3 and e_4 satisfy relation rel_1, e_5 and e_3 satisfy rel_2, and e_5 and e_4 are related by rel_1 (rel_1, rel_2, and rel_3 are elements of Rel_2).

of entity e_2 (who plays the role of a listener)." The situations that pertain to the same context share the same sets E, Ro, and Rel.

The condition to move between two contexts is one of the following: E is replaced by a different set (e.g., the set E of participants is now replaced with the set E' of family members), Ro has changed (e.g., the roles of the speaker and the listener are replaced with that of parent), or Rel has changed (e.g., in addition to spatial relationships, temporal relationships between entities may now matter).

The condition to move between two situations is one of the following:

- The cardinality of the set E has changed. For example, two persons enter the room and are recognized by the system as participants (their observables match the characteristics and behavior of participants). The system may provide the two latecomers with a summary of the current talk. If the latecomers are recognized as the organizers of the conference, then the system would detect a context change (not a situation change), because a new role (i.e., that of an organizer) is coming into play.
- A role assignment to an entity has changed (e.g., participant e switches from speaker to listener).
- A relation between two entities has changed (e.g., participant e was in front of e'. Now, e' is in front of e).

The ontology does not specify the nature of the entities, roles, relations, and observables. These are abstract classes from which a domain-dependent model can be specified. Using expressions of observables, designers identify the set of entities, roles, and relations that are relevant to the case at hand. Reignier et al. (2007) use this ontology and these principles for moving between situations and contexts for the development of smart environments. For UI plasticity, the observables of a context of use are organized into three information spaces that model the user, the environment, and the computing platform.

52.2.1.2 Observables of the Context of Use

The observables of a context of use define three information spaces: (1) the user model, (2) the environment model, and (3) the platform model.

1. The user model denotes the attributes and functions that describe the archetypal person who is intended to use, or is actually using, the interactive system. This ranges from basic user preferences as provided by most interactive systems, to more sophisticated descriptions such as profiles, idiosyncrasies, and current activities inferred from the repetitive use of services, of commands sequences and current tasks.
2. The environment model includes attributes and functions that characterize the physical places and times where the interaction will take place or is actually taking place. As for the user model, the number of candidate dimensions is quite large. It includes numeric locations (e.g., GPS coordinates) and/or symbolic locations (e.g., at home, in a public space, on the move in the street, a train or a car), numeric and symbolic temporal characteristics (e.g., 4th of January vs. winter), social rules and activities, as well as physical human perceivable conditions such as light, heat, and sound (using numeric and/or symbolic representations).

3. The platform model describes the computing, sensing, networking, and interaction resources that bind together the physical environment with the digital world. In the conventional GUI paradigm, the platform is limited to a single computing device, typically a workstation or a smart phone, connected to a network, and equipped with a fixed set of interaction resources such as a screen, keyboard, and stylus. Technological advances have enabled individuals to assemble and mould their own interactive spaces from public hot spots and private devices to access services within the global computing fabric. Interactive spaces will soon take the form of autonomous computing islands, or ecosystems, whose horizon will evolve, split, and merge under human control. Resources will be coupled opportunistically to amplify human activities where any real-world object has the potential to play the role of an interaction resource. Among many others, the Siftables (Merrill, Kalanithi, and Maes 2007), the History Tablecloth (Gaver 2006), as well as Skinput (Harrison, Desney, and Morris 2010) illustrate this trend. As a result, the platform must be modeled as a dynamic cluster of heterogeneous resources, rather than as a conventional mono-computing static device.

52.2.2 Usability

The term "usability" is interpreted in different ways by authors, even within the same scientific community. Usability has been identified with ease of use and learning, while excluding utility (Shackel 1984; Nielsen 1993). In other cases, usability is used to denote ease of use and utility, while ignoring learning. In SE, usability is considered an intrinsic property of the software product, whereas in HCI, usability is contextual: a system is not intrinsically usable or unusable. Instead, usability arises relatively to the context of use.

The contextual nature of usability has been recently recognized by the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 9126 standards developed in the software community with the overarching notion of "quality in use." Quality in use is "the capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety, and satisfaction in specified contexts of use." Unfortunately, as shown in Figure 52.2, usability is viewed as one independent contribution to quality in use. Thus, the temptation is high for software people to assimilate usability to cosmetic issues limited to the UI component of a software product, forgetting that system latency, reliability, missing functions, and inappropriate sequencing of functions have a strong impact on the system "useworthiness."

Useworthiness is central to Cockton's argument for the development of systems that have value in the real world (Cockton 2004, 2005). In value-centered approaches, software design should start from the explicit expression of an intentional creation of value for a selected set of target contexts of use. Intended value for target contexts are then translated into evaluation criteria. Evaluation criteria are not necessarily elicited from generic intrinsic features such as time for task completion but are contextualized. They are monitored and measured in real usage to assess the achieved value.

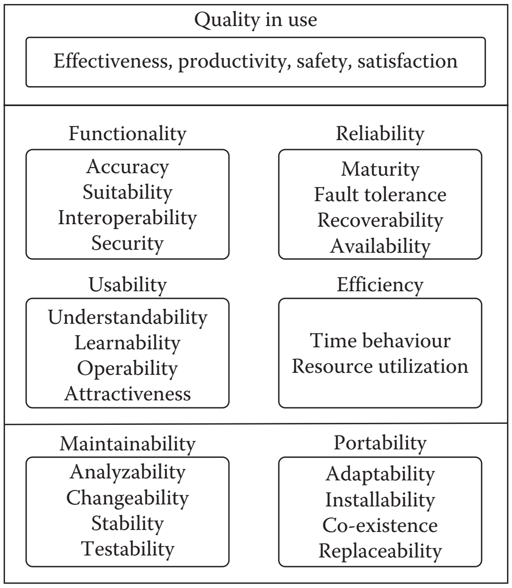


FIGURE 52.2 The usability model from ISO/IEC 9126-1.

Building on Cockton's approach, we suppose that for each of the target contexts of use C_i of a system, an intended value V_i has been defined and that V_i has been translated into the set of triples $\{(c_{i1}, d_{i1}, w_{i1}), \ldots, (c_{ij}, d_{ij}, w_{ij}), \ldots (c_{\text{in}}, d_{\text{in}}, w_{\text{in}})\}$, where c_{ij} is an evaluation criteria, and d_{ij} and w_{ij}, the expected domain of values and relative importance (the weight) of c_{ij} in C_i. As just discussed, c_{ij} may be a generic measurable feature or a customized measure that depends on the intended value in C_i. Usability U_i of the system for context C_i is evaluated against a combining function F_i on the set $\{(c_{i1}, d_{i1}, w_{i1}), \ldots, (c_{ij}, d_{ij}, w_{ij}), \ldots (c_{\text{in}}, d_{\text{in}}, w_{\text{in}})\}$ whose result is intended to lie within a domain of values D_i.

Coming back to the notion of plasticity, *an interactive system S is plastic from a source context of use C_i to a target context of use C_j* if the following two conditions are satisfied:

1. Adaptation, if needed, is supported when switching from C_i to C_j.
2. Usability (value) is preserved in C_j by the adaptation process. In other words, the usability function F_j defined for C_j lies within its intended domain D_j.

The domain of plasticity of a system is the set C of contexts of use C_i for which usability is achieved. We have defined usability by reasoning at the context level. If needed, a finer grain of reasoning can be applied at the situation level: intended value is defined for each situation of each context and then translated into evaluation criteria. Preserving usability is then evaluated on situation changes.

These definitions provide a theoretical framework where value comes first and is defined on a per-context (or situation) of use basis. For each of the intended target contexts (or situations), value is operationalized into a mix of generic and customized metrics. The difficulty is the identification of the relevant contexts of use and situations as well as the appropriate translation of value into significant metrics. We have no answer for operationalizing value, except to use generic measures when applicable, to instrument the system appropriately using sound software development techniques, such as aspect-oriented programming (Elrad, Filman, and Bader 2001), and to apply a healthy dose of common sense. However, our ontological framework on context and its associated method (Rey 2005) can be used to define the boundaries of contexts and situations of use as well as their relationships. For our notion of context of use, the fundamental entities are the user(s), environment, and platform, each of them being characterized by observables monitored by the system. Section 52.7 shows how to integrate the monitoring of observables within the software architecture of an interactive system.

52.3 PROBLEM SPACE OF USER INTERFACE PLASTICITY

Figure 52.3 captures the problem space of UI plasticity, where each branch denotes an issue along with the possible options for resolution. This problem space is characterized by (but not limited to) the following dimensions: the means used for adaptation (i.e., remolding and redistribution); the smallest UI units that can be adapted by the way of these means (from the whole UI considered a single piece of code to the finest grain: the interactor); the granularity of state recovery after adaptation has occurred (from the session level to the user's last action); the UI deployment (static or dynamic) as a way to characterize how much adaptation has been predefined at design time versus computed at runtime; the context coverage to denote the causes for adaptation with which the system is able to cope; the coverage of the technological spaces (TSs) as a way to characterize the degree of technical heterogeneity that the system supports; and the existence of a meta-UI to allow users to control and evaluate the adaptation process. A subset of these dimensions is now developed in detail.

52.3.1 Adaptation Means: UI Remolding and UI Redistribution

52.3.1.1 UI Remolding

UI remolding consists in changing the "shape" of the UI by applying one or several transformations on all, or parts, of the UI. These transformations include suppression of the UI components that become irrelevant in the new situation/context; insertion of new UI components to provide access to new services relevant in the new situation/context; substitution of UI components when UI components are replaced with new ones (substitution can be viewed as a combination of suppression and insertion); and reorganization of UI components by revisiting their spatial layout and/or their temporal dependency.

Remolding a UI from a source to a target UI may imply changes in the set of the available modalities. UI remolding is *intramodal* when the source UI components that need to be changed are retargeted within the same modality. Note that if the source UI is multimodal, then the target UI is multimodal as well: intramodal remolding does not provoke any loss in the set of modalities. Remolding is *intermodal* when the source UI components that need to be changed are retargeted into a different modality. Intermodal retargeting may engender a modality loss or a modality gain. Thus, a source multimodal UI may be retargeted into a monomodal UI, and conversely, a monomodal UI may be transformed into a multimodal UI. Remolding is *multimodal* when it uses a combination of intra- and intermodal transformations. For example, TERESA supports multimodal remolding between graphics and vocal modalities (Berti and Paternò 2005).

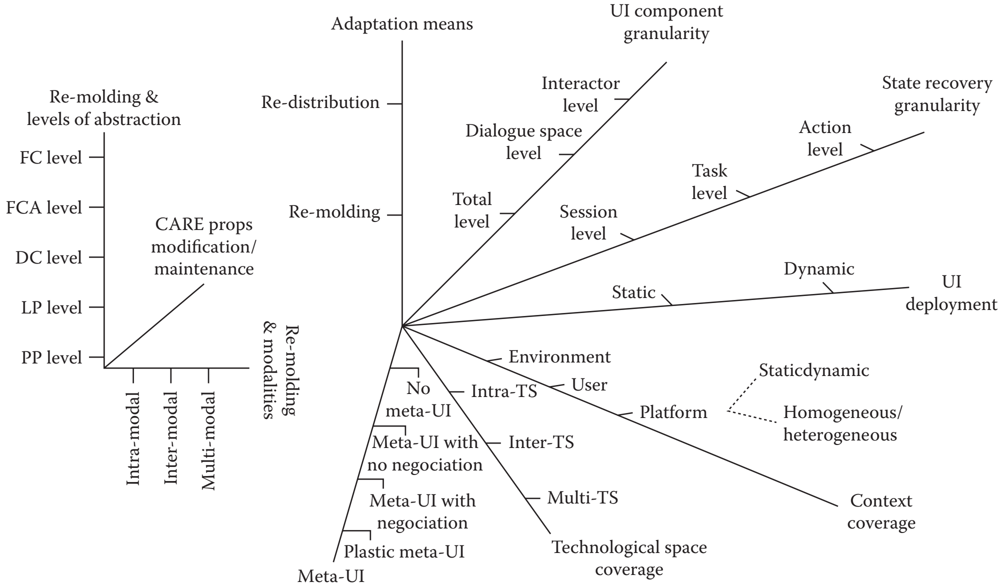


FIGURE 52.3 Problem space of user interface plasticity.

Remolding a UI may also change the way the CARE properties (Coutaz et al. 1995) are supported by the target UI (*complementarity*—several modalities must be combined to produce a semantically valid expression whether it be for input or output; *assignment*—only one modality can be used to produce a semantically valid input or output expression; *redundancy*—several equivalent modalities are used simultaneously to produce a semantically valid input or output expression; *equivalence*—several modalities are available to produce semantically equivalent expressions, but only one can be used at a time). Typically, because of a lack of computing power in the new situation, redundancy may be replaced with equivalence. Or, a synergistic complementarity (as in the "put-that-there" vocal sentence produced in parallel with two deictic gestures to denote "that" and "there") may be transformed into an alternate complementarity (as in the sequence: vocal "put that"; deictic gesture "that"; vocal "there"; and deictic gesture "there").

Transformations are performed at multiple levels of abstraction from cosmetic arrangements to deep software reconfiguration:

- At the physical presentation (PP) level, physical interactors (widgets) used for representing domain-dependent functions and concepts are kept unchanged, but their rendering and behavior may change. For example, if a concept is rendered as a button class, this concept is still represented as a button in the target UI. However, the look and feel of the button or its location in the workspace may vary. This type of adaptation is used in Tk as well as in Java/AWT with the notion of peers.
- At the logical presentation (LP) level, adaptation consists of changing the representation of domain-dependent functions and concepts. For example, the concept of month can be rendered as a Label1Textfield, or as a Label1Combobox, or as a dedicated physical interactor. In an LP adaptation, physical interactors can replace one another provided that their representational and interactional capabilities are equivalent. The implementation of an LP-level adaptation can usefully rely on the distinction between Abstract Interactive Objects and Concrete Interactive Objects as presented by Vanderdonckt and Bodart (1993). Changes at the LP level imply changes at the PP level.
- At the dialog component (DC) level, the tasks that can be executed with the system are kept unchanged, but their organization is modified. As a result, the structure of the dialog structure is changed. AVANTI's polymorphic tasks (Stephanidis and Savidis 2001) are an example of a DC-level adaptation. Changes at the DC level imply changes at the LP and PP levels.
- At the functional core adaptor (FCA) level, the nature of the entities as well as the functions exported by the functional core (which implements the domain-dependent concepts and functions) are changed. Changes at the FCA level imply changes at the DC, LP, and PP levels.

UI adaptation is often assimilated to UI remolding. This is true as long as we live in a closed world where the interaction resources are limited to that of a single computer at a time. In ubiquitous computing, the platform may be a dynamic cluster composed of multiple interconnected computing devices. In this kind of situation, instead of being *centralized*, the UI may be *distributed* across the interaction resources of the cluster.

52.3.1.2 UI Redistribution

UI redistribution denotes the reallocation of the UI components of the system to different interaction resources. The granularity of UI redistribution may vary from application level to pixel level:

- At the application level, the UI is fully replicated on each computing device. When the redistribution is dynamic, the whole UI of the application *migrates* to a new computing device, which in turn may trigger remolding.
- At the workspace level, the unit for distribution is the workspace. A workspace is a logical space that supports the execution of a set of logically connected tasks. This concept is similar to the notion of focus area used in contextual design for expressing the user-environment design. PebblesDraw (Myers 2001) and Rekimoto's Pick and Drop (Rekimoto 1997) are examples of UI distribution at the workspace level.
- The interactor level distribution is a special case of the workspace level in which the unit for distribution is an elementary interactor.
- At the pixel level, any UI component can be partitioned across multiple resources. For example, in the seminal smart room DynaWall (Streitz et al. 1999), a window may simultaneously lie over two contiguous white boards as if these were managed by a single computer.

52.3.2 State Recovery

The granularity of state recovery characterizes the effort users must apply to carry on their activity after adaptation has occurred. State recovery can be performed at the session, task, and physical action levels:

- When the system state is saved at the session level, users have to restart the interactive system from scratch. They rely on the state saved by the functional core before adaptation is taking place.
- At the task level, the user can pursue the job from the beginning of the current interrupted task (provided that the task is attainable in the retargeted system).

- At the physical action level, the user is able to carry on the current task at the exact point within the current task (provided that the task is attainable in the retargeted system).

52.3.3 Coverage of Technological Spaces

"A technological space is a working context with a set of associated concepts, body of knowledge, tools, required skills, and possibilities" (Kurtev 2002). Examples of technological spaces (TS) include documentware concerned with digital documents expressed in XML, dataware related to database systems, ontologyware, and so on. Most UIs are implemented within a single TS, such as Tcl/Tk, Swing, html. This homogeneity does not hold anymore for plastic UIs because redistribution to different computing devices may require crossing TSs. For example, a Java-based UI must be transformed into WML when migrating from a personal digital assistant (PDA) to a WAP-enabled mobile phone.

TS coverage denotes the capacity of the underlying infrastructure to support UI plasticity across TSs: *Intra-TS* corresponds to UIs that are implemented and adapted within a single TS. *Inter-TS* corresponds to the situation where the source UI, which is expressed in a single TS, is transformed into a single distinct target TS. *Multi-TS* is the flexible situation where the source and/or the target UIs are expressed in distinct TSs as supported by the Comet toolkit (Demeure, Calvary, and Koninx 2008).

52.3.4 Existence of a Meta-UI (or Supra-UI)

A meta-UI (or a Supra-UI) is a special kind of end-user development environment whose set of functions is necessary and sufficient to control and evaluate the state of an interactive ambient space (Coutaz 2006). This set is *meta* (or *supra*) because it serves as an umbrella *beyond* the domain-dependent services that support human activities in this space. It is *UI*-oriented because its role is to allow users to control and evaluate the state of the ambient interactive space. By analogy, a meta-UI is to ambient computing what desktops and shells are to conventional workstations.

A *meta-UI without negotiation* makes observable the state of the adaptation process but does not allow the user to intervene. A *meta-UI incorporates negotiation* when, for example, it cannot make sound decisions between multiple forms of adaptation, or when the user must fully control the outcome of the process.

The balance between system autonomy and too many negotiation steps is an open question. Another issue is the *plasticity of the meta-UI* itself. Thus, the recursive dimension of the meta-UI calls for the definition of a *native bootstrap meta-UI* capable of instantiating the appropriate meta-UI as the system is launched. This is yet another research issue.

The following examples illustrate the problem space of plastic UIs.

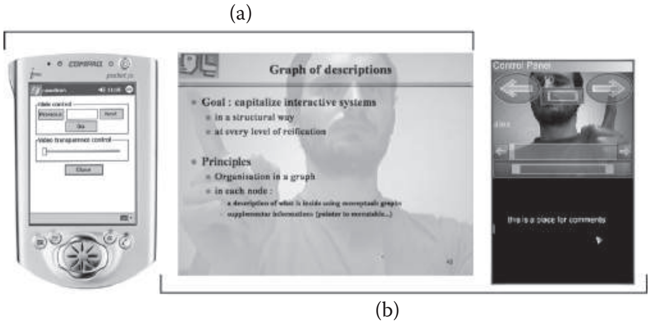
52.4 Case Studies

CamNote and Sedan-Bouillon are two examples of plastic interactive systems developed according to the MDE approach presented next. The services they provide are accessible from different types of computing devices including workstations, PDAs, and mobile phones. The UI components of these systems can be dynamically distributed and migrated across the interaction resources currently available in the interactive space. CamNote and Sedan-Bouillon differ in the TSs used for implementation: CamNote is Java centric, whereas Sedan-Bouillon uses PHP-MySQL Internet solutions. Whereas CamNote and Sedan-Bouillon offer a WIMP UI, the UI of our third example, Photo-Browser, includes a post-WIMP UI component. Photo-Browser has been developed to show how runtime adaptation can combine MDE with a code-centric approach.

52.4.1 CamNote

CamNote (for CAMELEON Note) is a slides viewer that runs on a dynamic heterogeneous platform. This platform may range from a single PC to a cluster composed of a PC and a PDA. Its UI is structured into four workspaces: (1) a slides viewer, (2) a note editor for associating comments to the slides, (3) a video viewer, also known as "mirror pixels," that shows a live video of the speaker, and (4) a control panel to browse the slides and to setup the level of transparency of the mirror. Speakers can point at items on the slide using their finger. This means of pointing is far more compelling and engaging than the conventional mouse pointer that no one can see. (Technically, the mirror is combined with the slides viewer using alpha-blending. See http://iihm.imag.fr/demos/CamNote/camnote_short.mov for a short movie demo.)

Figure 52.4a shows a configuration in which the graphical UI is distributed across the screens of a PC and PDA. The slide viewer is displayed in a rotative canvas so that it can be oriented appropriately when projected onto a horizontal surface. If the PDA disappears, the control panel automatically migrates to the PC screen. Because different resources are now available, the control panel includes different widgets and also a miniature representation of the speaker's video is now available. During the adaptation process, users



Graph of descriptions

- Goal : capitalize interactive systems
  - in a structural way
  - at every level of reification
- Principles
  - Organisation in a graph
  - in each node :
    - a description of what is inside using [illegible] graphs
    - supplementar informations (pointer to metatable...)

FIGURE 52.4 The user interface of CamNote. (a) The UI of CamNote when distributed on a PC and a PocketPC screen. (b) The control panel when displayed on the PC screen.

can see the control panel emerging progressively from the slides viewer, so that they can evaluate the progress of the adaptation process. The UI, which was distributed on a PC and a PDA, is now centralized on the PC (Figure 52.4b). Conversely, if the PDA reenters the interactive space, the UI automatically switches to the configuration of Figure 52.4a, and the control panel disappears from the PC screen by weaving itself into the slides viewer before reappearing on the PDA.

In this exemplar, context of use is limited to the platform. Transitions between situations occur at the arrival or departure of a computing device. Adaptation is based on redistribution of UI components at the workspace level. In turn, this redistribution triggers an intramodal GUI remolding at the dialog controller level: when the control panel resides on the PDA, the note-editing task is no longer available. Adaptation is automatic: the user has no control over the adaptation process, but a minimum of meta-UI exists (i.e., the weaving effect) to express the transition between two situations. State recovery is performed at the physical action level: the slides show is not disturbed by adaptation.

52.4.2 Sedan-Bouillon Website

Sedan-Bouillon is a website that aims at promoting tourism in the regions of Sedan (France) and Bouillon (Belgium) (http://www.bouillon-sedan.com/). It provides tourists with information for visiting and sojourning in these regions including a selection of hotels, camping, and restaurants. Figure 52.5a shows a simplified version of this website when a user is logged in from a PC workstation.

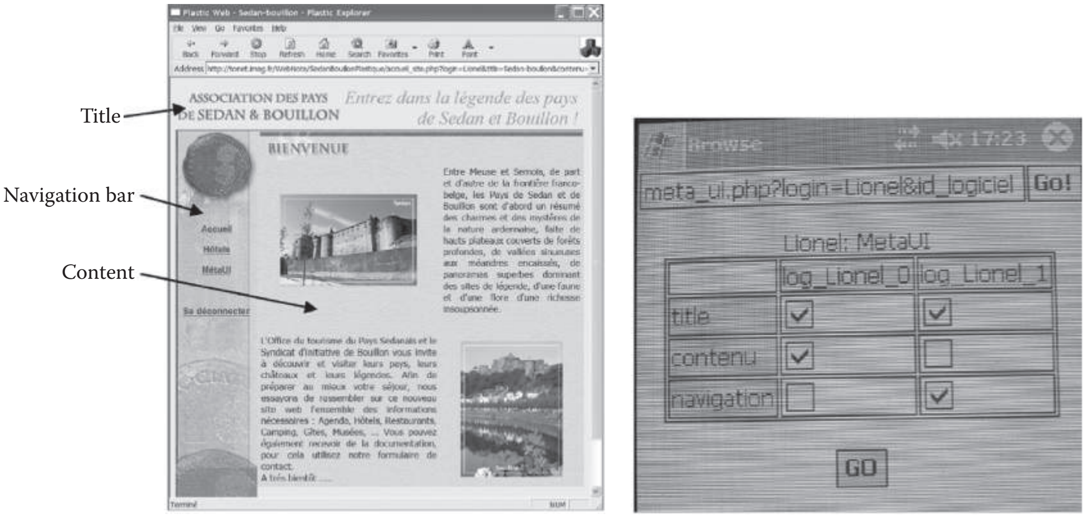
Preparing a trip for vacation is an exciting experience when shared by a group of people. However, one single PC screen does not necessarily favor collaborative exploration. By dynamically logging to the same website with a PDA, users are informed on the PDA that they can distribute the UI components of the site across the interaction resources currently available. In Figure 52.5b, the user asks for the following configuration: the title must appear on the PDA and the PC (the title slots are ticked for the two browsers available), whereas the content should stay on the PC and the navigation bar should migrate to the PDA. Figure 52.6 shows the resulting UI. At any time, the user can ask for a reconfiguration of the UI by selecting the "meta-UI" link in the navigation bar. The UI will be reconfigured accordingly.

Within the problem space of UI plasticity, the Sedan-Bouillon website is very similar to CamNote: same model of context of use, adaptation based on redistribution at the workspace level, with GUI intramodal remolding at the workspace level. Contrary to CamNote, remolding is performed at the LP level (no task is suppressed or restructured), and state recovery is supported at the task level: if adaptation occurs as the user is filling a form, the content of the form is lost by the adaptation process. Contrary to CamNote, the user has full control over the reconfiguration of the UI using the control panel provided by the meta-UI.

52.4.3 Photo-Browser

Photo-Browser supports photo browsing in a centralized or distributed way depending on the availability of a dynamic set of heterogeneous devices. These include a DiamondTouch interactive table, a wall, and a smart phone running Windows, MacOS X, and Android. The UI of Photo-Browser is dynamically composed of the following:

- A Tcl-Tk component running on the multipoint interactive surface (Figure 52.7a)
- A Java component that shows a list of the image names (Figure 52.7b)


Title
Navigation bar
Content
Plastic Web - Sedan-bouillon - Plastic Explorer
File View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites Print Font
Address http://lionel.imag.fr/WebNoms/SedanBouillonPlastique/accueil_site.php?login=Lionel&ttt=Sedan-bouillon&contenu
Accueil
Hôtels
MétaUI
Se déconnecter
Terminé
NUM
Browse
17:23
Go!
GO

ASSOCIATION DES PAYS DE SEDAN & BOUILLON

Entrez dans la légende des pays de Sedan et Bouillon !

BIENVENUE

Entre Meuse et Semois, de part et d'autre de la frontière franco-belge, les Pays de Sedan et de Bouillon sont d'abord un résumé des charmes et des mystères de la nature ardennaise, faite de hauts plateaux couverts de forêts profondes, de vallées sinueuses aux méandres encaissés, de panoramas superbes dominant des sites de légende, d'une faune et d'une flore d'une richesse insoupçonnée.

L'Office du tourisme du Pays Sedanais et le Syndicat d'initiative de Bouillon vous invite à découvrir et visiter leurs pays, leurs châteaux et leurs légendes. Afin de préparer au mieux votre séjour, nous essayons de rassembler sur ce nouveau site web l'ensemble des informations nécessaires : Agenda, Hôtels, Restaurants, Camping, Gîtes, Musées, ... Vous pouvez également recevoir de la documentation, pour cela utilisez notre formulaire de contact.
A très bientôt ......

(a)

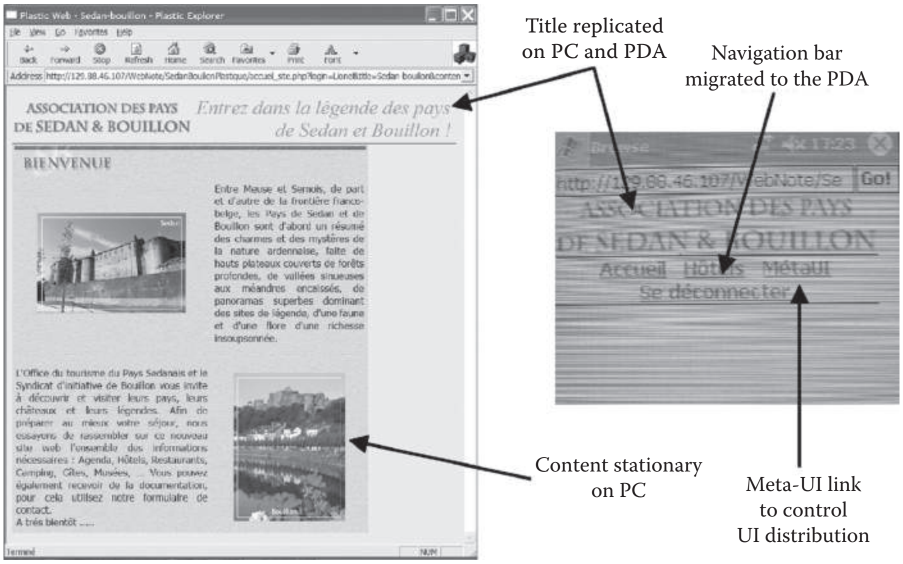
meta_ui.php?login=Lionel&id_logiciel

Lionel: MetaUI

	log_Lionel_0	log_Lionel_1
title	☑	☑
contenu	☑	□
navigation	□	☑

(b)

FIGURE 52.5 The Sedan-Bouillon website. (a) UI centralized on a PC screen. (b) The control panel of the meta-UI to distribute UI workspaces across the resources of the interactive space. The lines of the matrix correspond to the workspaces, and the columns denote the browsers currently used by the same user.


Plastic Web - Sedan-bouillon - Plastic Explorer
File View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites Print Font
Address http://129.88.46.107/WebNote/SedanBouillonPlastique/accueil_site.php?login=Lionel&title=Sedan bouillon&conten
Terminé
NUM
Title replicated on PC and PDA
Navigation bar migrated to the PDA
Browse
http://129.88.46.107/WebNote/Se
Go!
Accueil Hôtels MétaUI
Se déconnecter
Content stationary on PC
Meta-UI link to control UI distribution

ASSOCIATION DES PAYS DE SEDAN & BOUILLON

Entrez dans la légende des pays de Sedan et Bouillon !

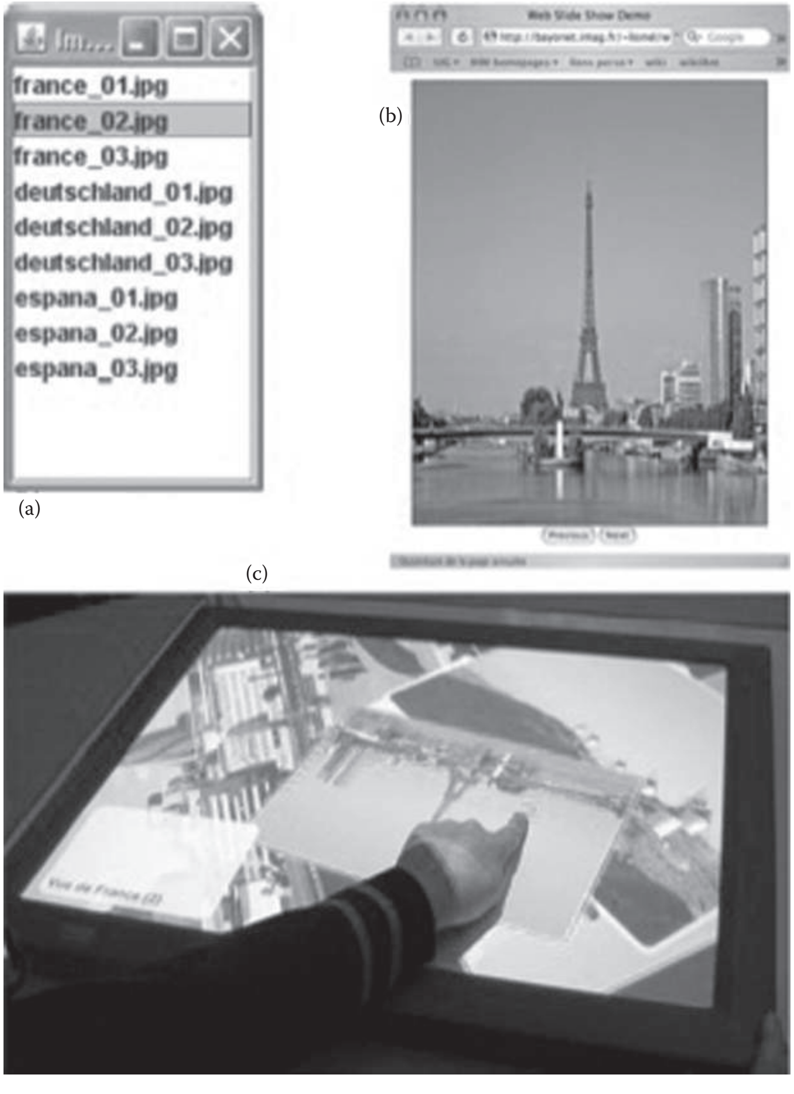
BIENVENUE

Entre Meuse et Semois, de part et d'autre de la frontière franco-belge, les Pays de Sedan et de Bouillon sont d'abord un résumé des charmes et des mystères de la nature ardennaise, faite de hauts plateaux couverts de forêts profondes, de vallées sinueuses aux méandres encaissés, de panoramas superbes dominant des sites de légende, d'une faune et d'une flore d'une richesse insoupsonnée.

L'Office du tourisme du Pays Sedanais et le Syndicat d'initiative de Bouillon vous invite à découvrir et visiter leurs pays, leurs châteaux et leurs légendes. Afin de préparer au mieux votre séjour, nous essayons de rassembler sur ce nouveau site web l'ensemble des informations nécessaires : Agenda, Hôtels, Restaurants, Camping, Gîtes, Musées, ... Vous pouvez également recevoir de la documentation, pour cela utilisez notre formulaire de contact.
A très bientôt ......

ASSOCIATION DES PAYS DE SEDAN & BOUILLON

FIGURE 52.6 The Sedan-Bouillon website when distributed across the resources of the interactive space. The meta-UI link allows users to return to the configuration panel shown in Figure 52.5b.



france_01.jpg
france_02.jpg
france_03.jpg
deutschland_01.jpg
deutschland_02.jpg
deutschland_03.jpg
espana_01.jpg
espana_02.jpg
espana_03.jpg

FIGURE 52.7 The Photo-browser application: a dynamic composition of executable and transformable components, managed by a dynamic set of interconnected factories running on different platforms. (a) Windows, (b) MacOS X, and (c) Android.

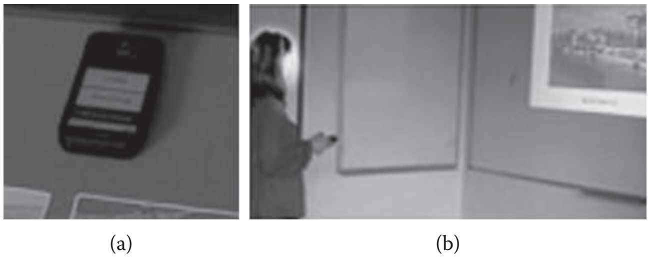


FIGURE 52.8 (a) Connecting a gPhone to the interactive space by laying it down on the interactive table. (b) Using the gPhone as a remote-controller to browse photos displayed by the HTML UI component of Figure 52.7c and video-projected on the wall.

- An HTML-based browser to navigate through the images set (Figure 52.7c)
- A Java component running on the gPhone to navigate sequentially through the photos using Next and Previous buttons (Figure 52.8)

The gPhone is dynamically connected to the interactive space by laying it down on the interactive table (Figure 52.8, left). As part of the platform, the gPhone can be used as a remote controller to browse photos displayed by the HTML UI component of Figure 52.7c and video-projected on the wall.

Within the problem space of UI plasticity, the context of use covered by Photo-Browser is a dynamic heterogeneous platform, and adaptation is multi-TS based on redistribution at the interactor level (i.e., photos) with no remolding. In its current implementation, the meta-UI is simulated using the Wizard of Oz technique. This meta-UI includes the recognition of three gestures: (1) a "wipe" gesture that allows the user to command the migration of the current selected photo from the table to the wall, (2) a "wipe" gesture that commands the system to shut down the table, and (3) the contact of the gPhone with the DiamondTouch.

Having characterized three exemplars in the problem space of UI plasticity, we now consider the method and mechanisms necessary to support UI plasticity. Although we advocate a MDE approach, we will analyze its limitations and suggest improvements.

52.5 MODEL-DRIVEN ENGINEERING

The motivation for MDE is the integration of knowledge and techniques developed in SE using the notions of model, model transformation, and mapping as the key concepts. In the early days of computer science, software systems were simple programs written in assembly languages. In those days, a code-centric approach to software development was good enough, not to say unavoidable, to ensure a fine control over the use of computing resources. Over the years, the field has evolved into the development of distinct paradigms and application domains, leading to the emergence of multiple TSs. Today, TSs can no longer evolve in autarky. Most of them share challenges of increasing complexity, such as adaptation, to which they can only offer partial solutions. Thus, we are in a situation where concepts, approaches, skills, and solutions need to be combined to address common problems. This is where MDE comes into play. MDE aims at achieving integration by defining gateways between TSs using a model-based approach. The hypothesis is that models, meta-models, model transformations, and mappings are everything.

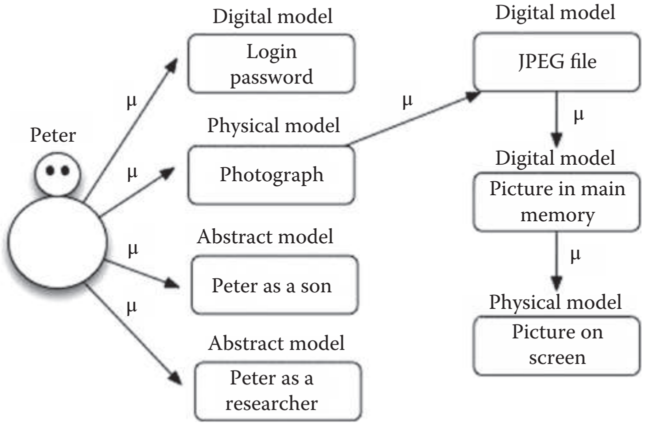


FIGURE 52.9 Models organized as oriented μ graphs (μ = "is represented by").

52.5.1 Models

A *model is a representation of a thing* (e.g., a system), *with a specific purpose*. It is "able to answer specific questions in place of the actual thing under study" (Bézivin 2004). Thus, a model, built to address one specific aspect of a problem, is by definition a simplification of the actual thing under study. For example, a task model is a simplified representation of some human activities (the actual thing under study), but it provides answers about how "representative users" proceed to reach specific goals.

A model may be *physical* (a tangible entity in the real world), *abstract* (an entity in the human mind), or *digital* (an entity within computers) (Favre 2004a,b). As illustrated in Figure 52.9, a printed photograph of a young man named Peter is a physical representation of Peter that his mother (for example) uses for a specific purpose. Peter's mother has mental representations of him as a good son or as a brilliant researcher (multiple abstract models about Peter). The authentication system that runs on Peter's computer knows him as a login name and password (digital model). If Peter's portrait is digitized as a JPEG picture, then the JPEG file is a digital model of a physical model. When displayed on the screen, the JPEG file is transformed into yet another digital graphics model in the system's main memory before being projected on the screen as an image (yet another physical model that Peter's mother can observe). As this example shows, models form oriented graphs (μ graphs) whose edges denote the μ relation "is represented by." In other words, a model can represent another model, and a model can be represented by several models.

Models may be *contemplative* (not able to be processed automatically by computers) or *productive* (able to be processed by computers). Typically, scenarios developed in

HCI (Rosson and Carroll 2002) are contemplative models of human experience in a specified setting. To be processed (by humans and/or by computers), a model must comply with some shared syntactic and semantic conventions: it must be a well-formed expression of a language. This is true for both productive and contemplative models; most contemplative models developed in HCI use a mix of drawings and natural language. A language is the set of all well-formed expressions that comply with a grammar (along with semantics). In turn, a grammar is a model from which we can produce well-formed expressions (or models). Because a grammar is a model of a set of models, it is called a "meta-model."

52.5.2 Meta-Model

A *meta-model is a model of a set of models that comply with it*. It sets the rules for producing models. It does not represent models. Models and meta-models form a tree: a model complies with a single meta-model, whereas a meta-model may have multiple compliant models.

As an example, suppose that the authentication system mentioned above is a Java program J. J is a digital model that represents Peter and that complies with the Java grammar G_J. G_J does not represent J, but defines the compliance of J with Java. G_J is one possible meta-model, but not the only one. The authentication system could also be implemented in C (yet another digital model of Peter). It would then be compliant with the C grammar G_C. Grammars G_C and G_J could be, in turn, produced from the same grammar such as Extended Backus-Naur Form (EBNF). EBNF, defined as the ISO/IEC 14977:1996 standard, is an example of a meta-meta-model, which is a model of a set of meta-models that are compliant with it. It does not represent meta-models, but sets the rules for producing distinct meta-models. As shown in Figure 52.10, the OMG model-driven architecture (MDA) initiative has introduced a four-layer modeling stack as a way to express the integration of a large diversity of standards using meta-object facility (MOF) as the unique meta-meta-model. MDA is a specific MDE deployment effort around industrial standards including MOF, UML, CWM, QVT. EBNF, G_J and G_C, and the Java and C programs are models that belong to the programming TS. Within the MDA TS, the java source code of our authentication system becomes a UML Java model compliant with the UML meta-model. In the XML Technological Space, the Java source code could be represented as a JavaML document compliant with a JavaML document type definition (DTD). (In the XML Technological Space, a DTD defines the legal building blocks of an XML document.)

As shown in Figure 52.10, the relation ("complies with") makes explicit the multiplicity of existing TSs and their systematic structure into three levels of modeling spaces (the so-called M1, M2, and M3 levels of MDA) plus the M0 level that corresponds to a system, or parts of a system. The μ and χ relations, however, do not tell how models are produced within a TS or how they relate to each other across distinct TSs. The notions of transformation and mapping are the MDE answer to this issue.

52.5.3 Transformations and Mappings

In the context of MDE, *a transformation is the production of a set of target models from a set of source models, according to a transformation definition*. A *transformation definition is a set of transformation rules* that together describe how source models are transformed into target models (Mens, Czarnecki, and VanGorp 2005). Source and target models are related by the τ relation "is transformed into." Note that a set of transformation rules is a model (a transformation model) that complies with a transformation meta-model.

Relation τ expresses an overall dependency between the source and target models. However, experience shows that finer grain of correspondence needs to be expressed. Typically, the incremental modification of one source element should be propagated easily into the corresponding target element(s) and vice versa. The need for traceability between the source and target models is expressed as mappings between source and target elements of these models. For example, as demonstrated in Section 52.6, the correspondence between a source task (and concepts) and its target workspace, window and widgets, is maintained as a mapping function.

Transformations can be characterized within a four-dimensional space:

- The *transformation may be automated* (it can be performed by a computer autonomously), semi-automated (requiring some human intervention), or manually performed by a human. For example, given our

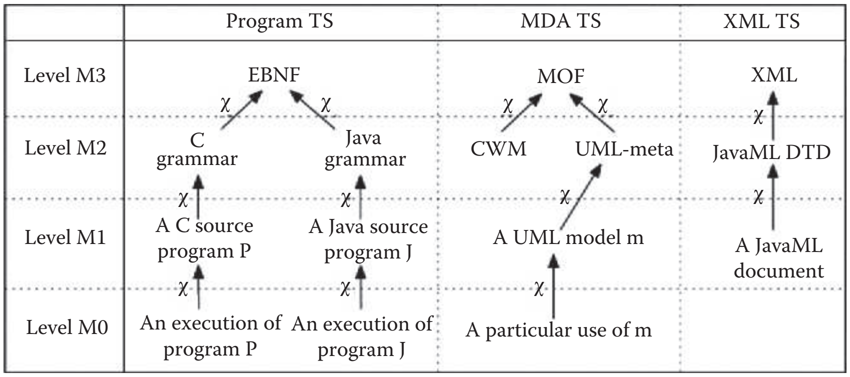


FIGURE 52.10 The OMG model-driven architecture four-layer stack.

current level of knowledge, the transformation of a "value-centered model" into a "usability model" can only be performed manually. On the other hand, UI generators such as CTTE (Mori, Paternò, and Santoro 2002, 2004) produce UIs automatically from a task model.

- A *transformation is vertical* when the source and target models reside at different levels of abstraction. UI generation is a vertical top-down transformation from high-level descriptions (such as a task model) to code generation. Reverse engineering is also a vertical transformation, but it proceeds bottom up, typically from executable code to some high-level representation by way of abstraction. A *transformation is horizontal* when the source and target models reside at the same level of abstraction. For example, translating a Java source code into C code preserves the original level of abstraction.
- Transformations are *endogenous* when the source and target models are expressed in the same language. They are *exogenous* when sources and targets are expressed in different languages while belonging to the same TS. For example, the transformation of a Java source code program into a C program is exogenous (cf. Figure 52.10).
- When *crossing TSs* (e.g., transforming a Java source code into a JavaML document), then additional tools (exporters or importers) are needed to bridge the gap between the spaces. Intertechnological transformations are key to knowledge and technical integration. This is the quest of MDE.

In Section 52.6, we show how the MDE principles have been applied to the development of plastic interactive systems by bringing together HCI practice and mainstream SE.

52.6 CONTRIBUTIONS OF MDE TO HCI AND PLASTIC USER INTERFACES

The HCI community has a long experience with models and meta-models, long before MDE existed as a field. In the 1980s, grammars (meta-models) were the formal basis for generating textual and graphical UIs (Hayes, Szekely, and Lerner 1985; Schulert, Rogers, and Hamilton 1985). MDE has helped the HCI community to define a shared vocabulary that expresses different perspectives on interactive systems as well as a reference framework for structuring the development process of plastic UIs.

52.6.1 Meta-Models as Different Perspectives on an Interactive System

Figure 52.11 shows an example of M2-level models that illustrate the principles of MDE applied to UI plasticity. These meta-models (and their relations) are intended to specify the canonic structures of the "important" concepts of the problem space of UI plasticity. These include, but are not limited to the following:

- M2-Tasks and M2-Concepts, respectively, define the notions of task and domain-dependent concepts. For example, in Figure 52.11, a task has a name and pre- and post-conditions. It may be composed of subtasks by the way of a binary operator (such as the AND, OR, SEQ operators) or decorated with a unary operator (such as Optionality, Criticity, and Default option).
- M2-Abstract UI (AUI) is a canonical expression of the rendering and manipulation of the domain-dependent concepts in a way that is independent of the concrete interactors (widgets) available on the target platform. It is expressed in terms of workspaces (as in Mara [Sottet, Calvary, and Favre 2006]), or in terms of Presentation Units (as in SEGUIA [Vanderdonckt and Bodart 1993; Vanderdonckt and Berquin 1999]), or in terms of Presentations (as in TERESA [Paternò 1999]). Workspaces, Presentation Units, and Presentations are synonyms to denote the same requirement: platform independence and absence of detailed UI design decisions.
- M2-Concrete UI (CUI) is an interactor-dependent expression of the UI. An interactor (e.g., a widget provided by an interaction toolkit such as Swing) is an entity of the UI that users can perceive (e.g., text, image, animation) and/or manipulate (e.g., a push button, a list box, a check box). A CUI expresses detailed UI design decisions.
- M2-Final UI (FUI) is the effective UI as perceived and is manipulated by the user at runtime. Typically, the same CUI Java code may behave differently depending on the Java virtual machine used at runtime.
- M2-Context of use is defined as a specialization of the ontology presented in "Context and Context of Use" where users, platforms, and physical environments are first-class entities.
- M2-Transformation supports the description of transformations that can be automated. Figure 52.12 shows an example of M1-level transformation using ATL as a meta-model to express the transformation of M2-Task compliant models into M2-Workspace compliant abstract UIs. Figure 52.13 illustrates the principles of this transformation graphically.

Considering the relations between the M2-level models of Figure 52.11, the BinaryOperators of M2-Task are related to navigation interactors to move between workspaces in the CUI. A task manipulates concepts (denoted by the "ConceptTask" relation). In turn, a concept is a class composed of a set of attributes. This class may inherit from other classes and may serve different purposes. A concept is represented as interactor(s)

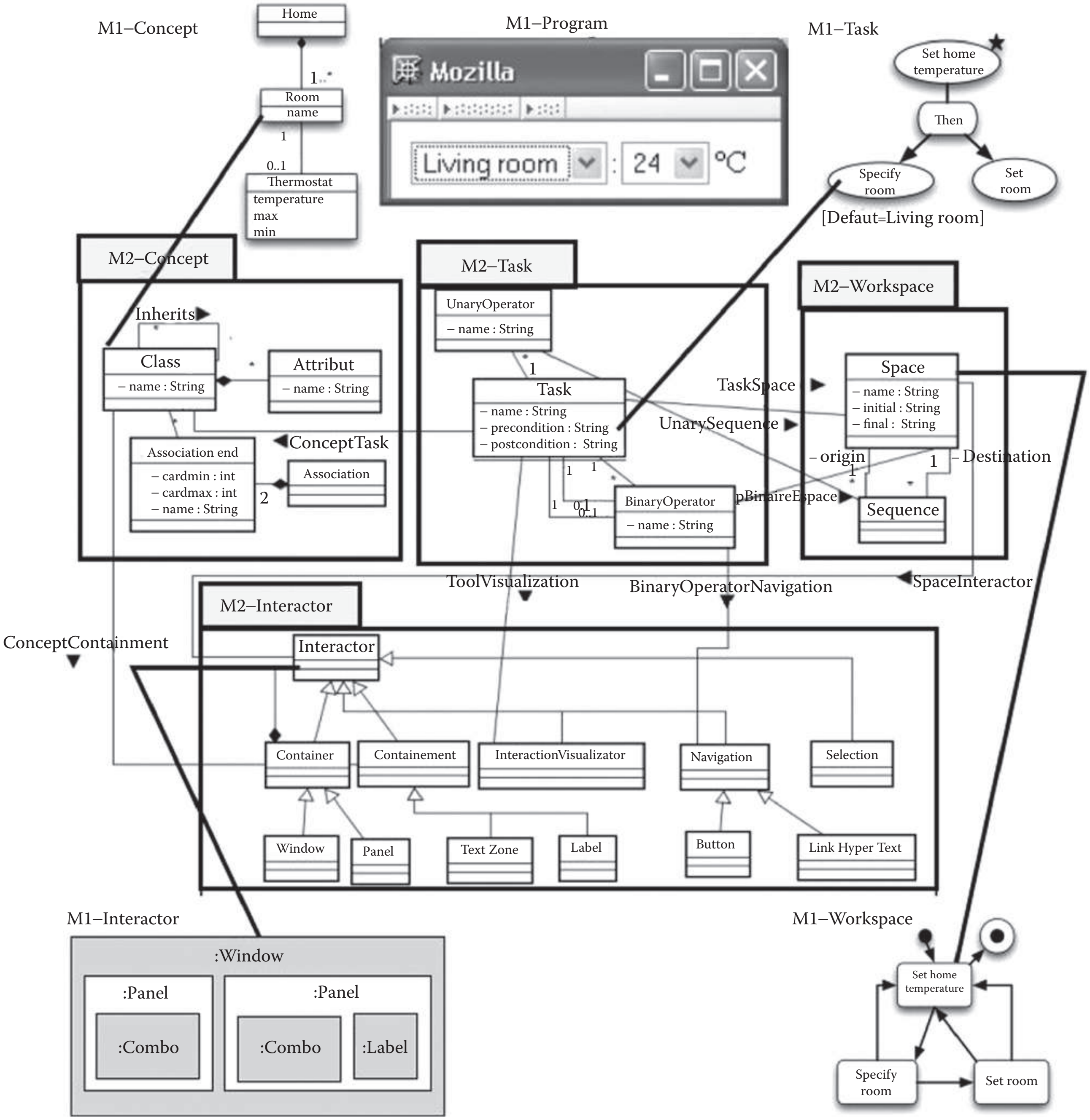
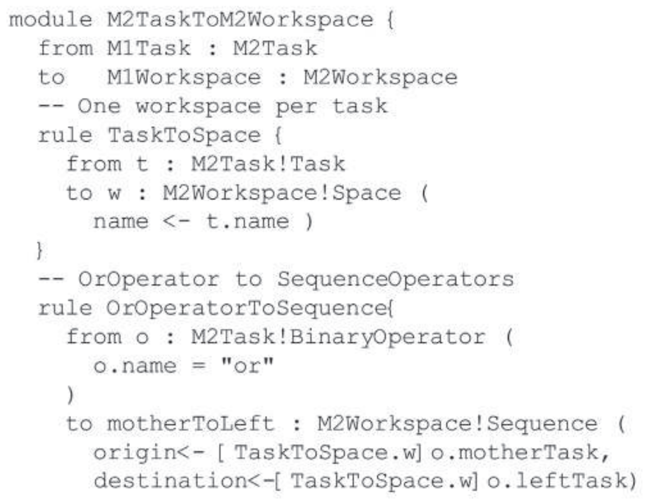


FIGURE 52.11 A home heating control system from an MDE perspective. A subset of the M1-level models and their mapping with their respective M2-level models. (For simplification purpose, only a subset of the mappings are represented.) (Adapted from Sottet, J.-S. 2008. *Méga-IHM: Malléabilité des Interfaces Homme-Machine Dirigée par les Modèles.* PhD Thesis, Université Joseph Fourier, Grenoble.)

in the CUI by the way of the ConceptContainment relation. The TaskSpace relation shows how a task relates to a workspace. M2-Workspace structures the task space at an abstract level, and M2-Interactor describes the interactors that will populate workspaces in the CUI. As shown by the definition of M2-Workspace, workspaces are chained with one another depending on the binary operator of the source tasks. A workspace is mapped into the CUI as a container class interactor. This container interactor, which, in the GUI modality, may be a panel or a window, is populated with interactors that render concepts and binary operations (navigation operators).

In addition to the examples of M2-level models, Figure 52.11 shows the M1-level models instantiated for a simple example: a home heating control system. At the top of Figure 52.11, M1-Task (which is compliant with M2-Task) has a name (“Set home temperature”), is repetitive (denoted by *), and is composed of two subtasks (“Specify room” and “Set Room”) that must be executed in sequence. By default, if “Specify room” is not executed, then the selected room is the living room. M1-Workspace (at the bottom right of Figure 52.11) is the AUI that corresponds to M1-Task. M1-Workspace is comprised of three workspaces (one per



```
module M2TaskToM2Workspace {
  from M1Task : M2Task
  to   M1Workspace : M2Workspace
  -- One workspace per task
  rule TaskToSpace {
    from t : M2Task!Task
    to w : M2Workspace!Space (
      name <- t.name )
  }
  -- OrOperator to SequenceOperators
  rule OrOperatorToSequence{
    from o : M2Task!BinaryOperator (
      o.name = "or"
    )
    to motherToLeft : M2Workspace!Sequence (
      origin<- [ TaskToSpace.w] o.motherTask,
      destination<-[ TaskToSpace.w] o.leftTask)
```

FIGURE 52.12 An ATL transformation description based on the meta-models shown in Figure 52.11. The rule *TaskToSpace* creates one workspace *w* per source task *t* where *w* takes the name of *t*; The rule *OrOperatorToSequence* transforms all OR operators *o* between two tasks (*o.leftTask* and *o.rightTask*) into two sequence operators (from *o.motherTask* to *o.leftTask*, and *o.leftTask* to *o.rightTask*).

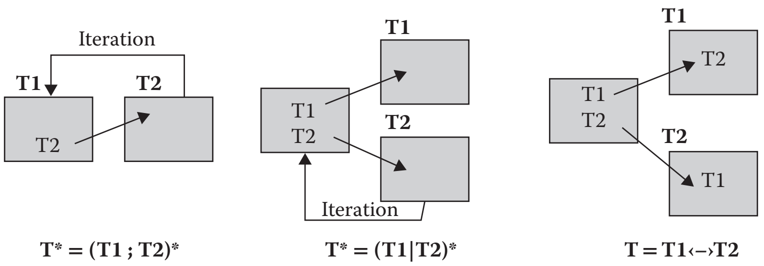


FIGURE 52.13 Typical transformation patterns between task models (expressed with UAN operators) and workspaces.

task) whose relations (denoted by arrows) express the navigation scheme between the workspaces. M1-Workspace is then populated with interactors to obtain a CUI. As shown at the bottom left of Figure 52.11, the "Set home temperature" workspace is mapped into a window, and the two others "Specify room" and "Set Room" become panels, populated with Combobox and Combobox with Label, respectively. At the top center of Figure 52.11, M1-Program represents the FUI of the home heating control system. All these M1-level models have been derived automatically by way of transformations.

Until recently, transformation rules were implemented as code within UI generators offering very little to no control over the resulting UI (Hayes, Szekely, and Lerner 1985; Schulert, Rogers, and Hamilton 1985). In addition, mappings were limited to the expression of correspondence (bindings) between elements of the UI with the API of the functional core (i.e., the business code). *MDE has helped the HCI community to promote transformation rules as models.* "Transformations as models" has three notable advantages—which, so far, have not been fully exploited by the HCI community: (1) It opens the way to knowledge capitalization and reuse: frequent transformations can serve as patterns in libraries, which in turn provide handles for intra- and inter-UI consistency. (2) Comparative evaluations of UIs can be performed in a controlled way, and UIs can be (re)targeted for different contexts of use using different transformations. (3) Most notably, transformations can be transformed, offering a powerful formal recursive mechanism for supporting UI plasticity. To our best knowledge, no research has been conducted on transforming transformations for UI plasticity. On the other hand, patterns are emerging (Taleb, Seffah, and Abran 2009), and early work has been initiated on UIs generated with different sets of transformation rules to support different usability criteria (Gajos, Wobbrock, and Weld 2008; Sottet et al. 2007).

The set of meta-models presented in Figure 52.11 is only one example among a plethora of proposals. Whereas there are a multiplicity of meta-models for task modeling (e.g., TeresaXML), the CUI level, on the other hand, is a very active area of research with no clear integrated vision. This is primarily due to the inherent diversity of interaction modalities and of interaction paradigms developed in HCI and also the diversity of TSs and of the competition among software vendors. To name just a few, the W3C recommendations include VoiceXML for voice integration, InkML for digital ink input representation with electronic pen, Extensible Multimodal Annotation Markup Language (EMMA) for multimodal input, not to mention 3D Markup Language (3DML) and Virtual Human Markup Language (VHML). Software vendors offer Macromedia Flex Markup Language (MXML), OpenLaszlo, XUL, XAML, and many other UI description languages (UIDL). As an answer to this plethora, the European ITEA2 UsiXML research project aims at covering the problem space of UI plasticity into a unified, systematic, and structured way with a clear motivation for standardization (http://usixml.org, http://itea.defimedia.be/usixml). The framework presented in Section 52.6.2 will serve as a basis for structuring the development process along with the appropriate tool support.

52.6.2 Reference Framework for a Structured Development Process

Over the years, the CAMELEON framework has progressively come to serve as a generic canonical structure for exploiting MDE to address the problem of UI plasticity (Calvary et al. 2003). In particular, the notion of transformation is used to cover many forms of the development process (see Figure 52.14). Typically in a forward engineering process, an AUI is derived from the domain-dependent concepts and task models. In turn, the AUI is transformed into a CUI, followed by the final executable UI. At the opposite, a reverse engineering process infers abstract models from more concrete ones using vertical bottom-up transformations. Translations may also be applied to transform a source model into a new model adapted for a different context of use.

Unlike the process initiated in the 1980s, which contained one entry point only at a high level of abstraction, the CAMELEON framework authorizes entry points at any level of abstraction from which any combination of horizontal and vertical bottom-up and top-down transformations can be applied. This theoretical flexibility means that the stakeholders involved in the development of an interactive system can use the development process that best suits their practice or the case at hand. In other words, the CAMELEON framework can be put in actions in many ways.

Seminal work in forward engineering for UI plasticity includes UIML (Phanariou 2000) and XIML (Puerta and Eisenstein 2001) that transform M1-level models into M0 level programs to support LP-level adaptation for centralized GUI. Tools for retargeting UIs such as Vaquita (Bouillon and Vanderdonckt 2002) and WebRevenge (Paganelli and Paternò 2003) correspond to a combination of bottom-up vertical, horizontal, and top-down vertical transformations. They lie within the same meta-meta level (the XML Technological Space), but they use distinct M2 meta-models. Vaquita and WebRevenge work offline. On the other hand, Digymes (Coninx et al. 2003) and Icrafter (Ponnekanti et al. 2001) generate CUI at runtime where a renderer dynamically computes a CUI from a workspace-level model expressed in XML. Websplitter (Han, Perret, and Naghshineh 2000) supports the distribution of web

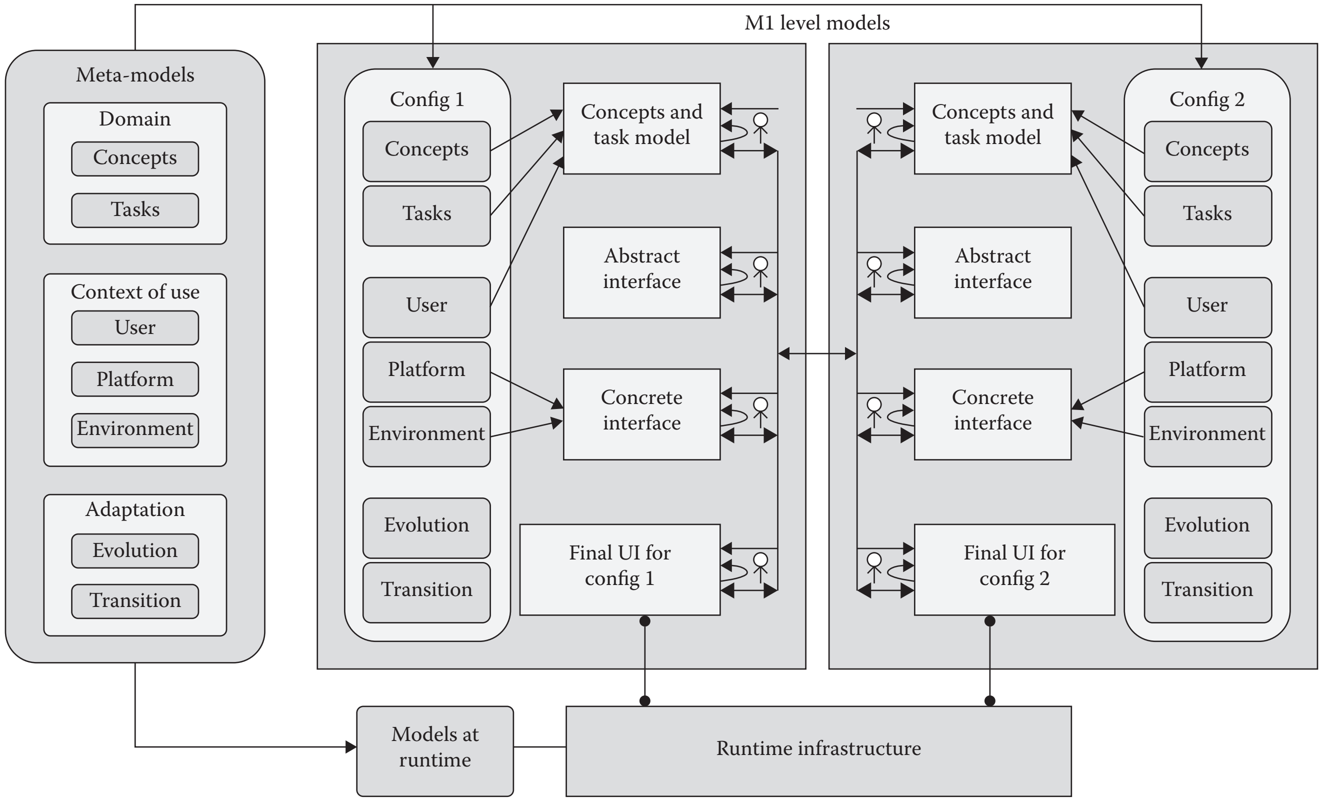


FIGURE 52.14 The CAMELEON reference framework for the development of plastic user interfaces.

pages content at the interactor level across the interaction resources of heterogeneous clusters, but distribution is statically specified in an XML policy file. As proof of concepts, small-size exemplars have been developed for different TSs.

To summarize, the CAMELEON reference framework is an MDE-compliant generic structuring conceptual tool for the development of plastic UIs:

- As a structuring reference framework, it federates the HCI community around a consensus.
- As a conceptual generic tool, it sets a vast agenda for technical research.
- As an MDE-compliant framework, it is still unclear in practice that formal modeling is the only way to go in HCI. This issue is discussed next.

52.6.3 Limits of MDE

The CAMELEON reference framework brings together the "right models," but the HCI community is far from having the "models right." As discussed above, the profusion of initiatives related to UIDL is symptomatic of the need and difficulty to define a coherent set of nonambiguous and easy to understand meta-models capable of covering the problem space of plastic UIs. In our opinion, two meta-models (at least) are key to the success of MDE for addressing UI plasticity: transformations and CUIs.

Transformations offer an elegant mechanism for full flexibility and technical integration. However, *transformations are hard to express*: QVT and ATL (Bézivin et al. 2003) are not languages for naive developers. In addition, usability rules (Sottet et al. 2007) are even harder to convey formally. More importantly, *inverse transformations cannot be automatically derived* for any source transformations. This is a fundamental flaw that may result in inconsistent models as transformations are performed up and down iteratively during the lifecycle of a system, breaking down the flexibility of the solution space envisioned by the CAMELEON reference framework. TransformiXML of the UsiXML (Limbourg 2004; Limbourg et al. 2004) meta-level environment, which is based on graphs transformations, is certainly a promising way.

At the CUI level, meta-modeling not only lags behind innovation but also bridles creativity. UIDLs for the expression of concrete UIs are technology driven instead of leaving rooms for new forms of interaction techniques. Although the CARE properties (Coutaz et al. 1995) have been devised 15 years ago, CUI languages have hardly scratched the surface of multimodal interaction. We are still unable to generate the paradigmatic "put-that-there" multimodal UI introduced more than 25 years ago (Bolt 1980). However, we do generate simplistic multimodal UIs based on XHTML+VoiceXML but with very limited microdialogs for interaction repair (Berti and Paternò 2005). Actually, CUI-level UIDLs are still struggling with the description of conventional GUIs for desktop computing. Meanwhile

- New forms of "constructable" computers such as the MIT siftables* and the CMU toy blocks† are put on the market.
- Novel interaction techniques are proliferating whether it be for supporting mobility (e.g., SixthSense [Mistry and Maes 2009]), for 3D interaction (where gesture and 3D screens are becoming predominant), or even for graphical tabletops and multisurface interaction (Balakrishnan and Baudisch 2009).
- New requirements are emerging: design is switching from the development of useful and usable systems for people with precise goals to engaging and inspired interaction spaces whose users can easily switch from consumers to creators.

In short, CUI meta-models need to capture the unbound vibrant convergence of physicality with "digitality." Perhaps, meta-modeling is, by essence, the wrong approach to CUIs: a model, which represents a thing, is necessarily a simplification, therefore a reduction, of the real thing. In these conditions, the subtle aspects of interaction, which make all the differences between constrained and inspired design, are better expressed using code directly in place of an abstraction of this code. However, this assertion should be mitigated by the following findings: designers excel at sketching pictures to specify concrete rendering. On the other hand, they find it difficult to express the dynamics, forcing them to use natural language (Myers et al. 2008). One way to fill the gap between designers' practice and productive models is to revive work à-la-Peridot (Myers 1990) such as SketchiXML (Coyette et al. 2004; Kieffer, Coyette, and Vanderdonckt 2010) where drawings are retro-engineered into machine-computable rendering. As for inferring behavior from examples, the promising "Watch What I Do" paradigm initiated in the late 1970s (cf. Dave Smith's Pygmalion system [Smith 1993]) is still an opened question.

In addition to impeding creativity, MDE, as a software development methodology, has favored the dichotomy between the design stages and the runtime phase, resulting in three major drawbacks:

1. Over time, models may get out of sync with the running code.
2. Design tools are intended for software professionals not for "the people." As a result, end users are doomed to consume what software designers have decided to be good for their hypothetic target users.
3. Runtime adaptation is limited to the changes of context identified as a key by the developers. Again, the envelope for end users' activities is constrained by design.

* http://sifteo.com/.

† http://www.modrobotics.com/.

Applied to UI development, the dichotomy between design and runtime phases means that UI generation from a task model cannot cope with ambient computing where task arrangement may be highly opportunistic and unpredictable. On the other hand, because the task model is not available at runtime, the links between the FUI and its original task model are lost. It is then difficult, not to say impossible, to articulate runtime adaptation based on semantically rich design-time descriptions. As a result, a FUI cannot be remolded beyond its cosmetic surface as supported by the CSS (Cascading Style Sheets).

Blurring the distinction between the design stage and the runtime phase is a promising approach. This idea is emerging in mainstream middleware (Ferry et al. 2009) as well as in HCI. The middleware community, however, does not necessarily address the concerns of the end user. Typically, a "sloppy" dynamic reconfiguration at the middleware level is good enough if it preserves system autonomy. It is not "observable" to the end user, whereas UI remolding and UI redistribution are! Thus, UI plasticity puts additional constraints on the developers and on the tools to support them. In particular, it becomes necessary to make explicit the transition between the source and the target UIs so that, in Norman's terms, end users can evaluate the new state. We need to pay attention to transition UIs in generic terms, not on a case-by-case basis.

In Section 52.7, we show how MDE can be reconciled with a code-centric approach at runtime.

52.7 MODELS AT RUNTIME

The combination of MDE with "code centricity" relies on three principles:

1. Principle 1: Cooperation between closed adaptiveness and open adaptiveness.
2. Principle 2: Runtime availability of high-level models.
3. Principle 3: Balance between the importance of Principles 1 and 2. The application of this principle is illustrated with Photo-Browser.

52.7.1 Principle 1: Cooperation between Closed Adaptiveness and Open Adaptiveness

"A system is open-adaptive if new application behaviors and adaptation plans can be introduced during runtime. A system is closed adaptive if it is self-contained and not able to support the addition of new behaviors" (Oreizy et al. 1999, page 55). By design, an interactive system has an "innate domain of plasticity": it is closed adaptive for the set of contexts of use for which this system/component can adapt on its own. For unplanned contexts of use, the system is forced to go beyond its domain of plasticity. It must be open adaptive so that a tier infrastructure (i.e., a middleware) can take over the adaptation process. The CAMELEON runtime conceptual architecture (CAMELEON-RT) shows how closed adaptiveness and open adaptiveness can be combined harmoniously (Balme et al. 2004). CAMELEON-RT shown in Figure 52.15 is a refinement of the box "Runtime infrastructure" at the bottom of Figure 52.14.

At the bottom of Figure 52.15, "Hardware" denotes a wide variety of physical entities: computing and communication facilities, interaction resources such as displays, mice, and stylus, as well as sensors and actuators. "Operating Systems" includes legacy OS such as Linux, MacOS, and Android; virtual machines such as the JVM; and modality interpreters such as speech and gesture recognition. Together, "Hardware" and "Operating Systems" constitute the ground basis of the interactive space, that is, the platform as defined in Section 52.2.

The top of Figure 52.15 shows the interactive systems (e.g., CamNote and Photo-Browser) that users are currently running in the interactive space. The meta-UI is one of them. A flower-like shape, ✿, denotes open adaptive components of these interactive systems. Components are open adaptative if they provide the world with management mechanisms. Management mechanisms include self-descriptive metadata (such as the current state and the services it supports and requires), and the methods to control its behavior such as start/stop and get/set-state. Software reflexivity coupled with a component model is a good approach to achieve open adaptiveness. The miniature adaptation-manager shape, ⊞⊡, denotes facilities embedded in the interactive system to support closed adaptiveness to observe the world, to detect situations that require adaptation, to compute a reaction that satisfies the new situation, and to perform adaptation. This functional decomposition is similar to that of the tier infrastructure shown in the center of Figure 52.15.

The tier infrastructure that supports open adaptiveness is structured in the following way:

- The context infrastructure builds and maintains a model of the context of use (Reignier et al. 2007). In turn, this infrastructure can be refined into multiple levels of abstraction, typically: raw data acquisition as numeric observables, transformation of raw data at the appropriate level of abstraction (e.g., as symbolic observables), which then feeds into situation management.
- The situation synthesizer computes the situation and possibly informs the evolution engine of the occurrence of a new situation. (This layer is in general considered as a part of the context infrastructure.)
- The evolution engine elaborates a reaction in response to the new situation.
- The adaptation producer implements the adaptation plan produced by the evolution engine. This is where the following dimensions of the problem space of UI plasticity come in play: granularity of UI remolding and/or redistribution, granularity of state recovery, coverage of TSs, and presence of a meta-UI.

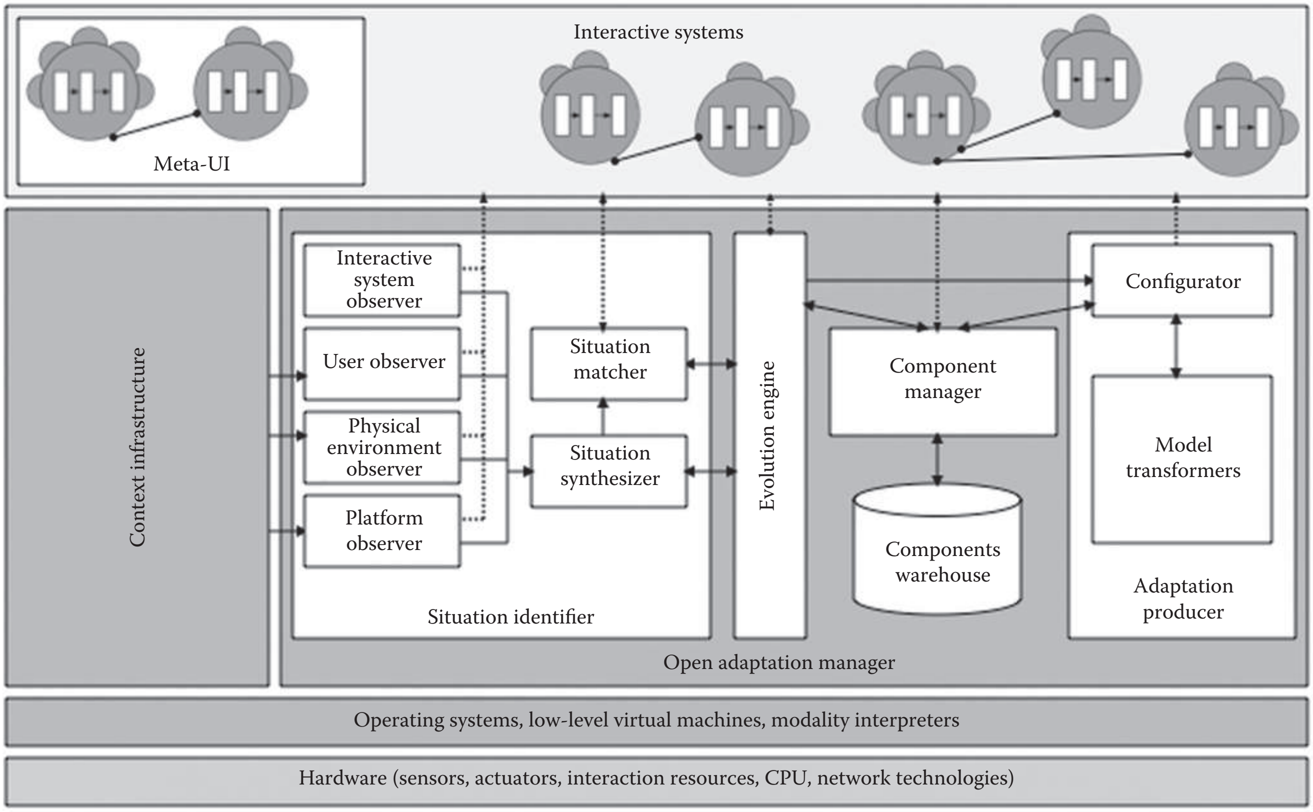


FIGURE 52.15 CAMELEON RT: a functional decomposition for supporting a mix of closed-adaptiveness and open-adaptiveness at runtime.

Such a functional decomposition is commonly used for the development of autonomic systems. To adapt this decomposition to plastic UIs, we propose the following improvements:

- The end user is kept in the loop: the reaction to a new situation may be a mix of specifications provided by developers or learned by the evolution engine based on observations of and reasoning on human and environmental behavior. In addition, the evolution engine as well as the adaptation producer may call upon end users' advice by the way of the meta-UI.
- The components referred to in the action plan do not necessarily exist as executable code. This is where Principle 2 comes into play.

52.7.2 Principle 2: Runtime Availability of High Level of Abstraction Models

At runtime, an interactive system is a set of graphs of models that express different aspects of the system at multiple levels of abstraction. As advocated by the CAMELEON framework, these models are related by mappings and transformations. As a result, an interactive system is not limited to a set of linked pieces of code. Models developed at design time, which convey high-level design decision, are still available at runtime for performing rational adaptation beyond cosmetic changes. When a component retrieved by the component manager is a high-level description such as a task model, the configurator relies on reificators to produce executable code as in Digymes (Coninx et al. 2003) and iCrafter (Ponnekanti et al. 2001). A retrieved component may be executable but may not fit the requirements. Ideally, it can be reversed engineered through abstractors, then transformed by translators and reified again into executable code (Bouillon and Vanderdonckt 2002).

52.7.3 Principle 3: Balance between Principles 1 and 2

By analogy with the slinky meta-model of the Arch model (Bass et al. 1992), the software developer can play with Principles 1 and 2. At one extreme, the interactive system may exist as one single-task model linked to one single AUI graph, linked to a single CUI graph, and so on (see Figure 52.16). This application of Principle 1 does not indeed leave much flexibility to cope with unpredictable situations unless it relies completely on the tier middleware infrastructure that can modify any of these models on the fly, then triggers the appropriate transformations to update the FUI. This approach works well for interactive systems for which conventional WIMP UIs are "good enough."

At the other extreme, the various perspectives of the system (task models, AUI, FUI, context model, etc.) as well as the adaptation mechanisms of the tier infrastructure are distributed across distinct UI service-oriented components,

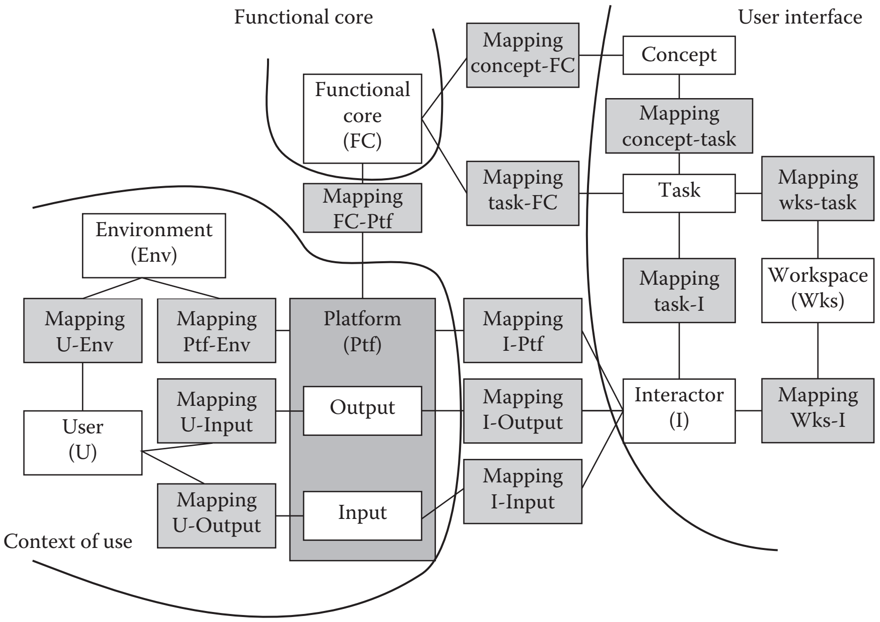


FIGURE 52.16 An interactive system as a graph of models available at runtime. These models are related by mappings and transformations.

each one covering a small-task grain that can be run in different contexts of use. This approach has been applied in the Comet toolkit (Demeure, Calvary, and Koninx 2008).

Basically, a comet is a plastic microinteractive system whose architecture pushes forward the separation of concerns advocated by PAC (Coutaz 1987) and MVC (Krasner and Pope 1988). The functional coverage of a comet is left open (from a plastic widget such as a control panel, to a complete system such as a powerpoint-like slide viewer). Each comet embeds its own task model, its own adaptation algorithm, as well as multiple CUIs and FUIs, each one adapted to a particular context of use. FUIs are hand coded possibly using different toolkits to satisfy our requirements for fine-grained personalization and heterogeneity. From the infrastructure point of view, a comet is a service that can be discovered, deployed, and integrated dynamically into the configuration that constitutes an interactive environment. The COTS (Bourguin, Lewandowski, and Tarby 2007), whose executable UI code is meta-described with the task they support, are based on similar ideas.

Figures 52.7 and 52.8 show another application of Principles 1 and 2 for the implementation of Photo-Browser. The FUI of Photo-Browser is dynamically composed of the following:

- A Tcl-Tk component running on a multipoint interactive surface (Figure 52.7a)
- A Java component that shows a list of the image names (Figure 52.7b)
- An HTML-based browser to navigate through the images set (Figure 52.7c)

Photo-Browser is implemented on top of a tier middleware infrastructure (called Ethylene) that covers the evolution engine, the component manager, and the adaptation producer of Figure 52.15. Ethylene is a distributed system composed of ethylene factories, each one running on possibly different processors (IP devices). The role of an ethylene factory is to manage the lifecycle of a set of components that reside on the same IP device as this factory and that have been registered to this factory. When residing on storage space, a component is meta-described using EthylenXML, an extension of the W3C standard WSDL (Web Service Definition Language). This meta-description includes the human task that the component supports, the resources it requires, and whether it is executable code or transformable code. In the latter case, it may be a task model, an AUI, a CUI, or even a graph of these models. For example, the HTML-based component (Fig 52.7c) is a CUI expressed in a variation of HTML. It must be transformed on the fly to be interpreted by an HTML renderer. The Tcl-Tk multipoint UI and the Java list are executable code. Their EthyleneXML meta-description specifies that they support image browsing and image selection tasks, that they need such and such interaction resources (e.g., a Tcl-Tk interpreter and a DiamondTouch interactive table) for proper execution, and that they require such communication protocol to be interconnected with other components. The gPhone UI component is an executable gPhone app that supports the next-previous browsing tasks (Figure 52.8). Interconnection between the components is initiated by the factories.

As shown by the examples above as well as by other works (Blumendorf, Leehmann, and S. Albayrak 2010; Clerckx, Vandervelpen, and Coninx 2007; Duarte and Carriço 2006; Savidis and Stephanidis 2010), the engineering community of HCI has focused its attention on runtime adaptation of the UI portion of an interactive system, not on the dynamic adaptation of the interactive system as a whole. The SE community is developing several approaches to enable dynamic bindings for service-oriented architectures. For example, Canfora et al. (2009) propose the dynamic composition of web services based on BPEL4People (that expresses a task-like model) as well as an extension of WSDL to meta-describe the services and using these two descriptions to generate the corresponding UI. Although bindings can be performed at runtime, users are confined within the workflow designed by the software developers. In addition, the generated UIs are limited to conventional WIMP UIs.

One promising approach to support flexibility at runtime is to consider the functional core components as well as UI components as services. In Ethylene, UI components adhere to this philosophy. They can be implemented in very different technologies, they can be discovered and recruited on the fly based on their meta-description, and they can be transformed on the fly. On the other hand, the business logic side of interactive systems is left opened. CRUISe (Pietschmann, Voigt, and MeiBner 2009) aims at supporting both sides in a uniform way but applies to the dynamic composition of web services and UI composition for the web (Yu et al. 2007).

52.8 CONCLUSION

An MDE has provided the HCI community with useful concepts for framing its own research agenda. Additional research is required for the definition of meta-models, transformations, and mappings provided that high-level descriptions can take full advantage of the latest innovations at the FUI level. Models at design time should not disappear at runtime but should be available to go beyond cosmetic adaptation. Design phase and runtime phase equal "même combat!"

Maximum flexibility and quality should be attainable by modeling the business logic as well as the UI as services with their own domain of plasticity. UI components should not be pure executable code. They have to be meta-described to express their exact nature and contracts with a human-centered perspective. They can be retrieved, transformed, and recomposed on the fly thanks to a tier middleware infrastructure. This middleware, which supports context, dynamic discovery, as well as the dynamic (re)composition of business logic and of transformable UI components, will permit interactive systems to go beyond their domain of plasticity. However, we must be careful at keeping the user in the loop while being able to produce transition UIs automatically.

The risk is that this wonderful apparatus will be designed for the specialists. We need to put the power in the people's hands and explore the potential from social programming. The success of the Apple App Store is a good indication for this. Mash-up tools have also started this trend for composing web-based applications (e.g., Google Gadgets or Yahoo! Widgets). More collaboration should be developed with the "cloud computing crowd." After all, an interactive space is a minicloud. If interaction resources were virtualized as memory, and network and computing resources are currently envisioned by the "systemers," then this would simplify enormously the development of UIs.

In short, MDE is an important tool for adaptation as long as it does not block creativity.

ACKNOWLEDGMENTS

This work is the result of research supported by the CAMELEON European project (IST-2000-28323) and by the Framework VI Network of Excellence SIMILAR. We thank the ANR CONTINUUM project (ANR-08-VERS-005) as well as the ITEA 08026 UsiXML project for supporting this research. We also thank Gaëtan Rey for the development of the contextor infrastructure, Lionel Balme and Alexandre Demeure for the implementation of CamNote, and the Sedan-Bouillon website as well as for the development of the first version of the runtime infrastructure for plastic UIs. Thanks to Jean-Marie Favre and Jean-Sébastien Sottet for their insights into MDE principles.

REFERENCES

Balakrishnan, R., and P. Baudisch. 2009. Special issue on ubiquitous multi-display environments. *Hum Comput Interact* 24 (1–2):1–8. Philadelphia, PA: Taylor & Francis.

Balme, L., A. Demeure, N. Barralon, J. Coutaz, and G. Calvary. 2004. CAMELEON-RT: A software architecture reference model for distributed, migratable, and plastic user interfaces, lecture notes in computer science. In *Ambient Intelligence: Second European Symposium, EUSAI 2004*, Vol. 3295, ed. P. Markopoulos, B. Eggen, E. Aarts et al., 291–302. Eindhoven, the Netherlands: Springer-Verlag Heidelberg, ISBN: 3-540-23721-6.

Bass, L., R. Faneuf, R. Little, N. Mayer, B. Pellegrino, S. Reed, R. Seacord, S. Sheppard, and M. Szczur. 1992. Arch, a metamodel for the runtime architecture of an interactive system. The UIMS developers workshop. *SIGCHI Bull* 24(1):32–7. ACM Publ.

Berti, S., and F. Paternò. 2005. Migratory multimodal interfaces in multidevice environments. In *Proceedings International Conference on Multimodal Interfaces (ICMI 05)*, 92–9. New York: ACM.

Bézivin, J. 2004. In search of a basic principle for model driven engineering. *European Journal of the Informatics Professional* 5(2):21–24.

Bézivin, J., G. Dupé, F. Jouault, G. Pitette, and J. Rougui. 2003. First experiments with the ATL transformation language: Transforming XSLT into Xquery. In 2nd OOPSLA Workshop on *Generative Techniques in the context of Model Driven Architecture, Anaheim, CA, USA*. http://www.softmetaware.com/oopsla2003/mda-workshop.html.

Blumendorf, M., G. Leehmann, and S. Albayrak. 2010. Bridging models and systems at runtime to build adaptive user interfaces. In *Proc. of the 2010 ACM SIGCHI Symposium on Engineering Interactive Computing Systems, EICS 2010*, 9–18. New York: ACM .

Bolt, R. 1980. Put that there": Voice and gesture at the graphics interface. In *Proc. of the 7th International Conf. on Computer Graphics and Interactive Techniques*, 262–70. New York: ACM.

Bouillon, L., and J. Vanderdonckt. 2002. Retargeting web pages to other computing platforms. In *Proceedings of IEEE 9th Working Conference on Reverse Engineering WCRE'2002 (Richmond, 29 October–1 November 2002)*, 339–48. Los Alamitos, CA: IEEE Computer Society Press.

Bourguin, G., A. Lewandowski, and J.-C. Tarby. 2007. Defining task oriented component. In *Proc. TAMODIA 2007*, Lecture Notes in Computer Science 4849. 170–83. Berlin: Springer.

Calvary, G., J. Coutaz, D. Thevenin, Q. Limbourg, N. Souchon, L. Bouillon, and J. Vanderdonckt. 2003. A unifying reference framework for multi-target user interfaces. *Interact Comput* 15(3):289–308. Elsevier Science B.V.

Canfora, G., M. Di Penta, P. Lombardi, and M. L. Villani. 2009. Dynamic composition of web applications in human centered processes. In *PESOS '09 Proceedings of the 2009 ICSE Workshop on Principles of Engineering Service Oriented Systems*, 50–57. Washington, DC: IEEE Computer Society.

Clerckx, T., C. Vandervelpen, and K. Coninx. 2007. Task-based design and runtime support for multimodal user interface distribution. In *Proc. of Engineering Interactive* Systems, Lecture Notes in Computer Science, LNCS 4940, 89–105. Berlin: Springer.

Cockton, G. 2004. From quality in use to value in the world. In *ACM Proceedings CHI 2004*, Late Breaking Results, 1287–90. New York: ACM.

Cockton, G. 2005. A development framework for value-centred design. In *ACM Proceedings CHI 2005*. Late Breaking Results, 1292–5. New York: ACM.

Coninx, K., K. Luyten, C. Vandervelpen, J. Van den Bergh, and B. Creemers. 2003. Dygimes: Dynamically generating interfaces for mobile computing devices and embedded Systems. In *Proceedings of the 5th International Symposium, Mobile HCI*, Lecture Notes in Computer Science, LNCS 2795, 256–70. Berlin: Springer.

Coutaz, J. 1987. PAC, an implemention model for dialog design. In *Proceedings of Interact'87*, 431–6. Stuttgart North-Holland: Amsterdam.

Coutaz, J. 2006. Meta user interfaces for ambient spaces. In *Proc. TAMODIA 2006, 5th International Workshop on Task Models and Diagrams for User Interface Design TAMODIA'2006*. Lecture Notes in Computer Science, LNCS 4385, Berlin: Springer.

Coutaz, J., J. Crowley, S. Dobson, and D. Garlan. 2005. Context is key. *Commun ACM* 48(3):49–53. ACM Publ.

Coutaz, J., L. Nigay, D. Salber, A. Blandford, J. May, and R. Young. 1995. Four easy pieces for assessing the usability of multimodal interaction: The CARE properties. In *Proceedings of the INTERACT'95*, 115–20. Chapman&Hall Publ.

Coyette, A., S. Faulkner, M. Kolp, Q. Limbourg, and J. Vanderdonckt. 2004. SketchiXML: Towards a multi-agent design tool for sketching user interfaces based on USIXML. In *Proceedings of the 3rd Annual Conference on Task Models and Diagrams, TAMODIA 2004*. Prague, Czech Republic. New York: ACM.

Demeure, A., G. Calvary, K. Koninx. 2008. A software architecture style and an interactors toolkit for plastic user interfaces. In *Proceeding of the 15th International Workshop DSV-IS 2008, LNCS*, 225–37. Berlin: Springer.

Dey, A. K. 2001. Understanding and using context. *J Pers Ubiquitous Comput* 5:4–7. Springer London.

Dourish, P. 2001. *Where the Action Is: The Foundation of Embodied Interaction*. Cambridge, MA: MIT Press.

Duarte, C., and L. Carriço. 2006. A conceptual framework for developing adaptive multimodal applications. In *Proc. of the 11th International Conference on Intelligent User Interfaces, IUI'06*, 132–9. New York: ACM.

Elrad, T., R. Filman, and A. Bader. 2001. Aspect oriented programming. Special issue. *Commun ACM* 44(10):28–95.

Favre, J. M. 2004a. *Foundations of Model (Driven) (Reverse) Engineering*. Dagsthul Seminar on Language Engineering for Model Driven Development, DROPS. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany http://drops.dagstuhl.de/portals/04101.

Favre, J. M. 2004b. *Foundations of the Meta-Pyramids: Languages and Meta-Models*. DROPS. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany http://drops.dagstuhl.de/portals/ 04101.

Ferry, N., G. Hourdin, S. Lavirotte, G. Rey, J.-Y. Tigli, and M. Riveill. 2009. Models at runtime: Service for device composition and adaptation. In *4th International Workshop Models@run.time, Models 2009 (MRT'09)* http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-509/MRT09_proceedings.pdf#page=55.

Gajos, K., J. Wobbrock, and D. Weld. 2008. Improving the performance of motor-impaired users with automatically-gen erated, ability-based interfaces. In *CHI '08: Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 1257–66. New York: ACM.

Gaver, W., J. Bowers, A. Boucher, S. Pennington, and N. Villar. 2006. The History tablecloth: Illuminating domestic activity. In *Proceedings of the 6th Conference on Designing Interactive Systems*, 199–208. New York: ACM.

Han, R., V. Perret, and M. Naghshineh. 2000. WebSplitter: A unified XML framework for multi-device collaborative web browsing. In *ACM Conference on Computer Supported Cooperative Work (CSCW 2000)*, 221–30. New York: ACM.

Harrison, C., T. Desney, and D. Morris. 2010. Skinput: Appropriating the body as an input surface. In *Proceedings of CHI'10, the 28th International Conference on Human Factors in Computing Systems*, 453–62. New York: ACM.

Hartson, R., A. Siochi, and D. Hix. 1990. The UAN: A user-oriented representation for direct manipulation interface designs. *ACM Trans Inf Syst (TOIS)* 8(3):181–203.

Hayes, P. J., P. Szekely, and R. A. Lerner. 1985. Design alternatives for user interface management systems based on experience with COUSIN. In *Proceedings of the ACM Conference on Human Factors in Computing Systems CHI'85*, 169–75. San Francisco, CA. New York: ACM.

Kieffer, S., A. Coyette, and J. Vanderdonckt. 2010. User interface design by sketching: A complexity analysis of widget representations. In *Proc. of the 2010 New York: ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, 57–66. New York: ACM.

Krasner, G. E., and S. T. Pope. 1988. A cookbook for using the model-view-controller user interface paradigm in smalltalk-80. *J Object Oriented Program (JOOP)* 1(3):26–49.

Kurtev, I., J. Bézivin, and M. Aksit. 2002. Technological spaces: An initial appraisal. In *International Conference on Cooperative Information Systems (CoopIS), DOA'2002 Federated Conferences, Industrial Track*, 30 Oct–1 Nov 2002, Irvine, CA. 1–6. http://eprints.eemcs.utwente.nl/10206/.

Limbourg, Q. 2004. *Multi-Path Development of User Interfaces*. Belgium: PhD of University of Louvain La Neuve.

Limbourg, Q., J. Vanderdonckt, B. Michotte, L. Bouillon, and V. Lopez-Jaquero. 2004. UsiXML: A language supporting multi-path development of user interfaces. In *Proceedings of 9th IFIP Working Conference on Engineering for Human-Computer Interaction Jointly with 11th Int. Workshop on Design, Specification, and Verification of Interactive Systems, EHCI-DSVIS'2004*, Lecture Notes in Computer Science, LNCS 3425. Hamburg, Germany. Berlin: Springer.

Mens, T., K. Czarnecki, and P. Van Gorp. 2005. A taxonomy or model transformations. In *Dagstuhl Seminar Proceedings 04101*. http://drops.dagstuhl.de/opus/volltexte/2005/11 Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

Merrill, D., J. Kalanithi, and P. Maes. 2007. Siftables: Towards sensor network user interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction*, TEI 2007. New York: ACM.

Mistry, P., and P. Maes. 2009. SixthSense—A wearable gestural interface. In *Proc. SIGGRAPH Asia 2009*. Yokohama, Japan: Emerging Technologies.

Mori, G., F. Paternò, and C. Santoro. 2002. CTTE: Support for developing and analyzing task models for interactive system design. *IEEE Trans Softw Eng* 28(8):797–813.

Mori, G., F. Paternò, and C. Santoro. 2004. Design and development of multidevice user interfaces through multiple logical descriptions. *IEEE Trans Softw Eng* 30(8):507–520.

Myers, B. 1990. Creating user interfaces using programming by example, visual programming, and constraints. *ACM Trans Program Lang Syst (TOPLAS)* 12(2):143–77. ACM Publ.

Myers, B. 2001. Using handhelds and PCs together. *Commun ACM* 44(11):34–41.

Myers, B., S. Y. Park, Y. Nakano, G. Mueller, and A. Ko. 2008. How designers design and program interactive behaviors. In *Proc. IEEE Symposium on Visual Languages and Human Centric Computing (VL/HCC)*, 177–84. IEEE Computer Society Press.

Nielsen, J. 1993. *Usability Engineering*. London: Academic Press.

Oreizy, P., M. Gorlick, R. Taylor, D. Heimbigner, G. Johnson, N. Medvidovic, A. Quilici, D. Rosenblum, and A. Wolf. 1999. An architecture-based approach to self-adaptive software. *IEEE Intell Syst* 14(3):54–62.

Paganelli, L., and F. Paternò. 2003. A tool for creating design models from website code. *Int J Softw Eng Knowl Eng* 13(2):169–89. World Scientific Publishing.

Paternò, F. 1999. *Model-Based Design and Evaluation of Interactive Applications*. Berlin: Springer Verlag.

Paternò, F. 2003. Concur task trees: An engineered notation for task models. In *The Handbook of Task Analysis for Human-Computer Interaction*, Chap. 24, ed. D. Diaper, and N. Stanton, 483–503. Mahwah, NJ: Lawrence Erlbaum Associates.

Phanariou, C. 2000. UIML: a Device-Independent User Interface Markup Language. PhD Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, september 2000.

Pietschmann, S., M. Voigt, and K. MeiBner. 2009. Dynamic composition of service-oriented web user interfaces. In *Proc. of the 4th International Conf. on Internet and Web Applications and Services, ICIW 2009*, 217–222. IEEE Computer Society Press.

Ponnekanti, S., B. Lee, A. Fox, P. Hanrahan, and T. Winograd. 2001. Icrafter: A service framework for ubiquitous computing environments. In *Proceedings Ubicomp 2001*, ed. G. Abowd, B. Brumitt, and S. Shafer, 57–75, LNCS 2201, Berlin: Springer.

Puerta, A., and J. Eisenstein. 2001. XIML: A common representation for interaction data. In *Proceedings IUI01*, 214–5. New York: ACM.

Reignier, P., O. Brdiczka, D. Vaufreydaz, J. L. Crolwey, and J. Maisonnasse. 2007. Context aware environments: From specification to implementation. *Expert Syst J Knowl Eng* 5(24):304–20.

Reignier, P., O. Brdiczka, D. Vaufreydaz, J. L. Crowley, and J. Maisonnasse. 2007. Contexte-aware environments: From specification to implementation. *Expert Syst J Knowl* Eng 24(5):305–20.

Rekimoto, J. 1997. Pick and drop: A direct manipulation technique for multiple computer environments. In *Proceedings of UIST97*, 31–9. New York: ACM Press.

Rey, G. 2005. *Le Contexte en Interaction Homme-Machine: Le Contexteur*. PhD Thesis. France: Université Joseph Fourier.

Rosson, M. B., and J. Carroll. 2002. *Usability Engineering Scenario-Based Development of Human Computer Interaction*. Burlington, MA: Morgan Kaufmann.

Savidis, A., and C. Stephanidis. 2010. Software refactoring process for adaptive user interface composition. In *Proc. of the 2010 ACM SIGCHI Symposium on Engineering Interactive Computing Systems, EICS 2010*, 19–28. New York: ACM.

Schulert, A. J., G. T. Rogers, and J. A. Hamilton. 1985. ADM-A dialogue manager. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'85)*, 177–83. San Francisco, CA. New York: ACM.

Shackel, B. 1984. The concept of usability. In *Visual Display Terminals: Usability Issues and Health Concerns*, ed. J. Bennett et al. Englewood Cliffs, NJ: Prentice-Hall, ISBN 0-13-942482-2.

Smith, D. C. 1993. Pygmalion: An executable electronic blackboard. In *"Watch What I Do", A. Cypher*, Chap 1. Cambridge, MA: MIT Press.

Sottet, J.-S. 2008. Méga-IHM: Malléabilité des Interfaces Homme-Machine Dirigée par les Modèles. PhD Thesis, Université Joseph Fourier, Grenoble.

Sottet, J.-S., G. Calvary, J. Coutaz, and J.-M. Favre. 2007. A model-driven engineering approach for the usability of user interfaces. In *Proc. Engineering Interactive Systems (EIS2007)*, ed. J. Gulliksen et al., 140–57. LNCS 4940. Berlin: Springer.

Sottet, J.-S., G. Calvary, and J.-M. Favre. 2006. *Models at Runtime for Sustaining user Interface Plasticity*. Int Models@run.time workshop, in conjunction with MODELS/UML.

Stephanidis, C., and A. Savidis. 2001. Universal access in the information society: Methods, tools, and interaction technologies. *J Univers Access Inf Soc UAIS* 1(1):40–55.

Streitz, N., J. Geibler, T. Holmer, S. Konomi, C. Müller-Tomfelde, W. Reischl, P. Rexroth, P. Seitz, and R. Steinmetz. 1999. i-LAND: An interactive landscape for creativity and innovation. In *Proceedings of the ACM Conference on Human Factors in Computer Human Interaction (CHI99)*, 120–7. New York: ACM.

Taleb, M., A. Seffah, and A. Abran. 2009. Interactive systems engineering: A pattern-oriented and model-driven architecture. In *Software Engineering Research and Practice*, 636–42. CSREA Press.

Thevenin, D., and J. Coutaz. 1999. Plasticity of user interfaces: Framework and research agenda. In *Proceedings Interact99*, ed. A. Sasse and C. Johnson, 110–7. Edinburgh: IFIP IOS Press.

Vanderdonckt, J., and P. Berquin. 1999. Towards a very large model-based approach for user interface development. In *Proc. of 1st Int. Workshop on User Interfaces to Data Intensive Systems UIDIS'99 (Edinburg, 5-6 September 1999)*, ed. N. W. Paton and T. Griffiths, 76–85. Los Alamitos, CA: IEEE Computer Society Press.

Vanderdonckt, J., and F. Bodard. 1993. Encapsulating knowledge for intelligent automatic interaction objects selection. In *Proceedings of the Joint ACM Conference on Human*

Factors in Computing Systems CHI and IFIP Conference on Human Computer Interaction INTERACT. Amsterdam, the Netherlands: ACM Press.

Van Lamsweerde, A. 2009. *Requirements Engineering: From System Goals to UML Models to Software Specifications*. 2009.

Winograd, T. 2001. Architectures for context. *Hum Comput Interact Spec Issue Context-Aware Comput* 16(2–4):401–20. Lawrence Erlbaum Associates.

Yu, J., B. Benatallah, R. Saint-Paul, F. Casati, F. M. Daniel, and M. Matera. 2007. A framework for rapid integration of presentation components. In *WWW'07 Proc. of the 16th International Conf. on World Wide Web*, 923–32. New York: ACM.

Part VI

The Development Process

Section C: Testing, Evaluation, and Technology Transfer

53 Usability Testing

Joseph S. Dumas and Jean E. Fox

CONTENTS

53.1 INTRODUCTION

At the turn of the millennium, a survey of usability professionals showed that they rated usability testing as the most influential method for having a strategic impact on organizations (Rosenbaum, Rohn, and Humburg 2000). At that time, testing frequently was recommended as a key to stimulating product development organizations to integrate user-centered design into the development process. It had strong face validity: it appeared to evaluate usability fairly, and tests always produced a list of usability problems to be addressed.

Testing's face validity and its value as a tool to influence developers delayed the profession's examination of the details of the method, its reliability, and more importantly its forms of validity. In the past decade, books and research studies have looked at both the strengths and limitations of the large variety of practices that are now part of the umbrella term "usability testing." In this chapter, we discuss those new materials. There are two themes that appear throughout: (1) the widespread use of Agile and other streamlined development practices has increased the pressure to test faster and cheaper and to strip testing of some of its essentials and (2) the lack of consensus about the criteria for what constitutes a valid usability test has made it vulnerable to attacks on what were assumed to be its basic foundations.

In a previous edition of this handbook, we focused primarily on the basic concepts of testing practice that were established over the period of its emergence and growth (Dumas and Fox 2007). In this edition, we focus on the body of research and opinion that has emerged during the past decade.

53.2 TYPES OF TESTS

The fact that the term "usability testing" refers to a wide variety of methods becomes apparent when one tries to categorize them. There are at least five dimensions to describe a particular test:

1. Purpose of the test—explore the usability of early design concepts, diagnose usability problems, fix usability problems, validate usability, measure baseline usability, or compare usability of products
2. Scope of the product tested—the whole product, part of it, and/or selected task flows
3. Location of sessions—local or remote
4. Presence of a test moderator—moderated or unmoderated
5. The level of functionality of the product—paper prototype, static screens, interactive prototype, or live code

We are not aware of any empirical data about the frequency of test types. We believe that the most common test, at this time, is a moderated diagnostic test on a subset of a product conducted locally in the middle of development. While the stated desire of many in the usability profession is to test earlier, it is not clear that early tests are most common, though testing has moved from the late stage method it was 20 years ago.

Alternative protocols are gaining in popularity. As we see in this chapter, remote and unmoderated online tests are more common. The rapid iterative test and evaluation (RITE) method is an example of a local, moderated test of a whole or part of a product, conducted early in development, with the purpose of fixing rather than finding problems.

In addition to this classification of types of usability tests, there are other terms that are used in the literature and in practice to describe tests: qualitative, quantitative, formal, and informal. What these terms denote is not always clear. They add to the ambiguity about what a usability test is.

53.3 TRADITIONAL DIAGNOSTIC USABILITY TEST

Over the past 20 years, the basic characteristics of a moderated, diagnostic usability test have been established:

- The focus is on usability. The traditional usability test is intended to uncover usability issues both positive and negative.
- The participants are end users or potential end users. Most usability professionals would agree that to have a valid diagnostic usability test, the participants must be part of the target market for the product. The key to finding people who are potential candidates for the test is a user profile (Branaghan 1997) or a persona (Pruitt and Adlin 2005). A user profile captures two types of characteristics: (1) those that the users share and (2) those that might make a difference among users. The test team must also determine how many participants per user group to include in the test. Five to eight users has become a common sample size.
- The participants perform tasks with a product or prototype, usually while thinking aloud. One of the essential requirements of every usability test is that the test participants attempt tasks that users of the product will perform. When a product of even modest complexity is tested, however, there are more tasks than there is time available to test them, so it is necessary to sample tasks. While not often recognized as a limitation of testing, the sample of tasks is a limitation to the scope of a test. Those components of a design that are not touched by the tasks the participants perform are not evaluated. Almost without exception, testers present the tasks that participants do in the form of a task scenario. For example, consider the following:

> You have just bought a new combination telephone and answering machine. The box is on the table. Take the

product out of the box and set it up so that you can make and receive calls.

- Before the test session starts, the administrator instructs the participant how the test will proceed and informs the participant that the test probes the usability of the product, not the participant's skills or experience. In most diagnostic usability tests, participants are asked to think aloud.
- The data are recorded and analyzed. In a usability test, there will be both quantitative and qualitative data. Quantitative data include measures of efficiency (e.g., task times), effectiveness (success rates), and satisfaction (ease-of-use ratings). Qualitative data include participant comments and tester observations. The data can be collected and recorded in a variety of ways. In the early days of usability testing, the test administrators recorded all data by hand with stopwatches and clipboards. Over the years, numerous tools have become available to automatically record video and data. Many of these tools also conduct basic data analysis, such as calculating average task times and success rates. Much of the data analysis involves building a case for a usability problem by combining several measures—a process that has been called "triangulation." In addition, problems are usually categorized by their severity.
- The results of the test are communicated to appropriate audiences. Test reporting began with lengthy written reports and highlight tapes, but reporting has become less formal.

53.4 UPDATING USABILITY TESTING BASICS

While many usability tests still are consistent with the traditional basics, the variations on what are still called "usability tests" have grown. In this section, we discuss how testing evolved.

53.4.1 From Usability to User Experience

Beginning about the year 2000, there was a concern that the "traditional" view of usability was limiting. These efforts have led the professional to ask whether task effectiveness, efficiency, and satisfaction are only part of the story. For example, Quesenbery (2004, 2005) broadened the ISO definition by adding e*ngaging:* "how pleasant, satisfying, or interesting an interface is to use" (Quesenbery 2004, p. 5). Others have advocated looking beyond traditional views of the scope of the profession to consider "user experience," shaped not only by usability, but by aesthetic, emotional, social, and business factors (Jordan 2002; Teague and Whitney 2002; Karat 2003; Hancock, Pepe, and Murphy 2005). Many industry groups have changed their name from "usability" to "user experience" groups.

This broadened view of what it takes for a product to be successful has had two important implications for usability practice. First, traditional usability measures are being adapted to assess the broader notions of user experience. Second, new methods are being used to supplement the more traditional ones (e.g., Karat 2003; Pagulayan et al. 2003; Murphy, Stanney, and Hancock 2003). Usability practitioners are supplementing traditional measures with value-based metrics and methods drawn from the marketing, anthropology, and psychology disciplines. Questions such as "Is it fun?," "Is it motivating?," and "Does it provide enough variety (as opposed to consistency)?" are a few examples of what usability practitioners are asking today in addition to "Is it usable?"

As a result of these changes, usability testers are including more subjective measures into tests, and testing is often paired with marketing methods such as online surveys to broaden the scope of the evaluation beyond usability issues.

53.4.2 Are Five Still Enough?

Part of the popularity of usability testing has come from its ability to find usability problems with only a few participants. Anyone who watches multiple test sessions with the same set of tasks perceives that the same issues begin to repeat, and that somewhere in the five-to-eight test participant range, with the same user population, it begins to seem unproductive to test more participants. So it was with great joy that testers greeted the research studies by Virzi (1990, 1992), showing that 80% of the total number of usability problems that will be uncovered by as many as 20 participants will be found by as few as five. Virzi also found that those five participants will uncover most of the problems judged by experts to be severe. This finding has been confirmed several times (Faulkner 2003; Law and Vanderheiden 2000). Practitioners conducting diagnostic, moderated tests continue to select small numbers of participants, confident that they are finding most of the problems that they could find.

Those findings lead to a popular rule of thumb for diagnostic tests that "five is enough." But the interpretation of the rule is not as simple as it appears. Among others, the rule has been attributed to Nielsen (2000). But Nielsen placed the rule into an iterative testing context in which he proposed that three iterative tests of the same product each with five participants are better than one test with 15 participants.

In Section 53.12, we discuss some recent studies showing that a single usability test only finds a small fraction of the total number that multiple independent tests will find. How do we reconcile that finding with the studies of sample size? All the studies that have looked at sample size and the number of problems found have done so with a single test by one test team. Apparently, there is a limitation in how many problems a single test team can find. At this point in time, we do not know why test teams have this limitation.

Furthermore, all the sample size studies, except Lewis (1994), tested very simple applications. As Redish (2007) points out, we know very little about the optimal usability testing process with complex systems. So the rule of thumb would be more accurate if it said that five participants will uncover about 80% of the problems that one team can find with a small application. That fact also means that adding more participants may not find more problems as long as the test team does not change.

There also have been a few challenges to the generality of the “five is enough” rule of thumb, most notably by Lewis (1994, 2001) and Turner, Lewis, and Nielsen (2006). Their challenge makes the reasonable case that tests differ in the probability of problem detection. A moderately complicated product being tested for the first time might indeed yield many of its problems with five to eight participants. Those authors have looked at problem detection over a large sample of tests and found that the average probability of detection is about 30%.

But what about a product that is being retested after most of its problems have been fixed? One might expect that it might take more participants because it is harder to detect the problems. It also may take more participants if the user population is very heterogeneous, such as with elderly and disabled users (Grossnickle 2004; ITTATC 2004; Swierenga and Guy 2003). Turner, Lewis, and Nielsen (2006) created and verified a formula for determining how many participants are needed in a variety of testing situations.

Finally, the pressure coming from organizations using an Agile development approach is to test with even fewer than five participants (see Section 53.7). Krug (2010) suggests monthly tests with three participants each. He argues that each test will find more than enough problems to keep the team busy for the next month. Again, Krug is saying that his rule needs to be viewed in an iterative testing context.

53.4.2.1 Sample Sizes with Other Testing Types

Most of the dialog about minimum sample size and all the research have been done in the context of diagnostic test with a moderator. The minimum sample size for comparison and baseline tests is much larger because of the need to measure usability not just to find problems. Minimum sample sizes for those types of tests are similar to those for cognitive science research studies, about 12–15 per group.

One of the strengths of online unmoderated testing is that much larger samples are easier to obtain. These larger samples can make the results of online tests more credible. By adding survey questions in addition to tasks, such tests can gather market research as well as usability data (Albert, Tullis, and Tedesco 2010).

53.4.3 Are “Real” Users Necessary?

The first books on testing procedures stressed that it is necessary to recruit test participants who are part of the target market for the product (Rubin 1994; Dumas and Redish 1993). The rationale was that all the problems that the target market will have would not be uncovered if a different population is tested. This rationale was based on a logical analysis and anecdotal evidence.

The methods for identifying the qualifications of participants were asking marketing experts in the organization, developing a user profile, or more recently, using personas (Pruitt and Adlin 2005). However, these methods result in ranges of qualifications that are difficult to cover with a small sample. For example, if one of the qualifications is knowledge of a software operating system, do you select participants with a little or a lot of experience? The advice is to make sure you have a range, some with a little and some with a lot. This strategy may mean that two subgroups are combined into one. Furthermore, as participants are recruited, compromises in the details of the qualifications are often made. Consequently, the final sample only approximates the profile or persona.

For usability testing, as with other types of research, it is nearly impossible to draw a random sample of the population. You may be limited by issues such as geography (e.g., those close enough to come to your lab), availability (e.g., who can participate during business hours), or willingness (e.g., who wants to participate). To some extent, every usability test sample is at least partly a sample of convenience. The challenge for testers is to determine which characteristics might affect the participants’ experiences with a product.

One of the consistent results of tests is that they always yield lists of problems, often long ones. It has seldom been necessary to question whether a different sample would have yielded a different list. But the pressure from Agile development and from startups to get websites to market faster has led to a practice called “hallway” testing (Spolsky 2000), in which “you grab the next person that passes by in the hallway and force them to try to use the code you just wrote. If you do this to five people, you will learn 95% of what there is to learn about usability problems in your code.” Krug (2010, p. 42) makes a similar point, “But there are many things you can learn by watching almost anyone use it (a website).”

Until a research study shows that a sample of “real” users, that is people who are part of the target market, yields the highest quality list of problems, some practitioners will continue to see value in recruiting a more convenient sample. As long as such samples uncover usability problems, it will be difficult to argue that the sample invalidates the test.

53.4.4 Does Task Selection Matter?

An essential component of any usability test is that participants attempt tasks. The measures taken during and after tasks provide the empirical data on which the product design is evaluated.

The selection of tasks is a function of the purpose and scope of the test. Testers must also consider the order of the tasks. In some cases, the tasks must be completed in a particular order, such as when a later task relies on the results of an earlier task or when there is a natural task order. In other tests, the order of tasks is randomized or varied in some way to balance any order or start up effects.

Task selection has been identified as one source of the lack of agreement in independent tests. Molich et al. (1998)

concluded that differences in usability test results across four teams were at least partially explained by fact that the teams use different tasks. However, Molich and Dumas (2008) found that even when teams used almost the same tasks, the problems they listed did not appear to have any more agreement than for teams with quite different task sets. This may have occurred because the task statement is only the starting point for the task. Participants can go down very different paths from the same starting point, thereby exposing different flaws.

In addition to the types of tasks to include, it is also important to consider the number of tasks. Lindgaard and Chattratichart (2007), using the same data as Molich and Dumas, found that the number of tasks used by teams was significantly correlated with the number of problems found, while the number of test participants recruited was not. Interestingly, they also found that the number of participants was not significantly correlated with either measure or the number of problems found. In this case, the number of tasks had greater influence on the number of problems found than on the number of participants.

Most of the advice about task selection and wording has been given in the context of moderated tests. The challenges of creating tasks for unmoderated online tests are quite a bit different (Albert, Tullis, and Tedesco 2010). Task statements for unmoderated tests must be clear and unambiguous because there is no moderator to clarify them. Careful piloting of wording is essential. "Easy to understand" is not the same as "easy to guess," as the participant may guess rather than perform the task. The best tasks are ones whose successful completion is obvious, such as an answer to a question that can be found on a web page. It may be necessary to constrain the participant in the path they use to complete a task to be sure the test is probing the product design appropriately. Finally, sometimes the participant must indicate whether they believe that they completed the task successfully. In such cases, an analysis of their path through the task may be needed to supplement their belief in their success.

53.4.5 Incorporating Thinking Aloud into Usability Testing

One of the early differences between a usability test and a research study was that the test participants typically thought aloud in a usability test. While concurrent thinking aloud is normally done as part of a diagnostic usability test, it is really a method of its own. It has been used in psychological research since the turn of the twentieth century, but it is best known as a cognitive psychology method for studying short-term memory (Ericsson and Simon 1993). Retrospective thinking aloud, that is thinking aloud while watching a video recording of task performance, is also used, especially in situations in which concurrent thinking aloud cannot or should not be done.

Concurrent thinking aloud provides usability testing with most of its drama. Without thinking aloud, it is unlikely that usability testing would have become the most influential usability method. It is the think aloud protocol that grabs the attention of first-time visitors to a usability test and gives a test session the appearance of a science-based method.

When usability testing was first being codified, thinking aloud was borrowed from cognitive psychology without much reflection. It was not until shortly after 2000 that usability specialists began to look at it more closely. Independently, Boren and Ramey (2000), Dumas (2001) and, more recently, Nielsen, Clemmensen, and Yssing (2002) went back to look more closely at what Ericsson and Simon (1993) had described and whether testing practitioners were really following that method. Those reviews showed that the descriptions of how to use the think aloud method that had been provided to usability testing practitioners by Dumas and Redish (1999) and Rubin (1994) were in direct contradiction to the instructions used in cognitive psychology research in which participants are discouraged from reporting feelings or expectations or to make any verbal diversions over and above the content of their actions. In usability testing, participants are encouraged to report on their feelings and expectations and on additional relevant issues.

Only a few research studies have been done on the think aloud method in a usability testing context. Krahmer and Ummelen (2004) compared typical usability testing think aloud instructions to the instructions used by Ericsson and Simon and found that the research instructions do not work well in a testing context. Ebling and John (2000) traced each usability problem found in a usability test back to its source in the test measures. They found that over half of the problems identified in their test came from the think aloud protocol alone. Their study supplements an earlier one by Virzi, Source, and Herbert (1993), who showed that fewer problems are identified when the participants do not think aloud. Eger et al. (2007) found that concurrent and retrospective think aloud protocols found approximately the same number of usability problems. However, when they included eye movements in the retrospective cue, they uncovered significantly more usability problems than in the traditional think aloud condition.

Two interesting questions about thinking aloud are "can everyone think aloud while performing another task?" and "should thinking aloud be done in all tests?" There are now a number of studies and demonstrations that suggest that many user populations cannot perform tasks and think aloud at the same time, including the following:

- Teen and preteen children (Als, Jensen, and Skov 2005)
- Low-literacy populations (Birru et al. 2004)
- People for whom English is not their first language (Evers 2004)
- People from some non-English speaking cultures (Evers 2004)

Evers (2004) conducted think aloud tests and post-test interviews with a sample of 130 high school students from England, North America, the Netherlands, and Japan.

The moderator was English. The Japanese students had the most difficulty with the think-aloud sessions. They felt uncomfortable speaking out loud about their thoughts and seemed to feel insecure because they could not confer with others to reach a common opinion. The English also needed reassurance before feeling comfortable with thinking out loud.

Concurrent thinking aloud also is to be avoided in tests of voice response system, tests using eye trackers (Bojko 2005), and tests that include complex tasks or complex environments (Redish and Scholtz 2007). van den Haak, de Jong, and Schellens (2003) found that participants performing complex tasks exposed fewer problems using concurrent thinking aloud than with retrospective thinking aloud.

Some authors have proposed alternatives to concurrent thinking aloud. Redish and Scholtz suggest using retrospective thinking aloud for testing complex and open-ended tasks. Als et al. used a technique with children called constructive interaction, in which children work in pairs on tasks. The pairs who used constructive interaction exposed more usability problems than the children who used thinking aloud. Strain, Shaikh, and Boardman (2007) conducted concurrent think aloud tests with blind participants and found the audio from the screen reader interfered with the conversation. Although the method worked when participants were familiar enough with the screen reader to pause and restart the audio easily, the authors suggest considering retrospective think aloud or what they call "Modified Stimulated Retrospective Think-Aloud." With this method, the participant walks through the application after completing the task.

Frøkjær and Hornbæk (2005) proposed a technique called "Cooperative Usability Testing" as a way to deal with the difficulties participants sometimes have with concurrent thinking aloud. In their technique, there are two parts to a test session. In the first part of the session, interaction, a test participant performs tasks while thinking aloud in the presence of an evaluator. The session is videotaped and the participant is allowed to ask questions of the evaluator, who takes a more active role than is typical. In the second part of the session, interpretation, the participant and one or more evaluators discuss the video of the interaction session with the goal of clarifying the usability problems. In their study, Frokjaer and Hornbaek report that evaluators and participants liked the cooperative technique and that it uncovered more problems. In addition, participants who just did a traditional think aloud session made negative comments about thinking aloud, including that it was hard to think aloud and perform difficult tasks or read text and that thinking aloud felt like "asocial" monolog. Participants also reported that what they were saying out loud was only a fraction of what they were thinking internally. Similarly, Eger et al. (2007) found that participants rated concurrent think aloud sessions as significantly more unpleasant and unnatural than a retrospective think aloud session. These studies are among the few to record comments about thinking aloud from the participant's point of view. We need more studies that provide data on what the thinking aloud experience is really like for test participants.

Studies of how thinking aloud instructions are actually given and how test moderators prompt participants to think aloud show that moderators are inconsistent (Boren and Ramey 2000; Norgaard and Hornbaek 2006). The think aloud method described in the books on testing techniques and in this section of the chapter is simply not followed in practice.

53.4.6 New Research on Testing Measures

Some recent studies have begun to clarify the relationships among the measures that are taken during tests. Testers have assumed that the measures should correlate. Usability problems often are identified through their impact on multiple measures. For example, a structural problem with the organization of a website might cause task failures, longer task times, errors, the need for assistance, and the participants rating tasks or the product as hard to use.

On the other hand, if the measures were highly correlated, testers would not need so many of them. Frøkjær, Hertzum, and Hornbæk (2000, p. 345) argued that, "Unless domain specific studies suggest otherwise, effectiveness, efficiency, and satisfaction should be considered independent aspects of usability and all be included in usability testing." Supporting that point, Hornbæk and Law (2007) reported weak correlations among efficiency, effectiveness, and satisfaction, with an average Pearson-product moment correlation (*r*) of about +0.2. The correlations were equally weak among time-on-task, completion rates, error rates, and user satisfaction. But many of the studies they analyzed were not usability tests.

Sauro and Lewis (2009) conducted an analysis of data from 90 summative usability tests conducted in industry settings. The pattern of correlations added some complexity to the discussion of whether measures do or do not correlate. They found that correlations among the performance measures were all significant and in the medium range, around or slightly higher than +0.5. The correlations between the performance measures and post-task ratings were slightly lower. Lowest of all, around +0.2, were correlations between post-test ratings and performance measures. Sauro and Lewis then performed a factor analysis on the correlations, which produced two factors: the first is heavily loaded with the three performance measures while the second is heavily loaded with the subjective measures. They argue that this pattern provides support for a construct of usability with a performance and subjective component and that using multiple measures increases the reliability of testing data.

53.4.6.1 Subjective Measures

Recent studies have begun to clarify several issues about subjective measures in usability testing. One of the issues is the format of rating scales. Tedesco and Tullis (2006) compared five different rating scale formats used for post-task ratings. They found that a simple five level Likert scale from very difficult to very easy was the most reliable. But none of the formats had acceptable reliabilities below sample sizes of

8–10 participants. Sauro and Dumas (2009) confirmed those findings and also found that a simple subjective mental effort scale performed as well as the Likert scale.

Tullis and Stetson (2004) conducted a similar study of post-test questionnaires, such as the System Usability Scale (SUS). They found that the 10-question SUS was the most reliable and that none of the questionnaires was reliable with sample sizes below 10 participants.

As discussed earlier, several studies have shown that correlations between post-test questionnaires and other measures are among the weakest. Sauro and Lewis (2009, p. 1617) notes: “It is reasonable to speculate that responses to post-test satisfaction questions elicit reactions to aspects beyond the immediate usability test (past usage, brand perception, customer support).”

For the practitioner, this research means that simple subjective measures are to be preferred, that none of these measures are reliable with the sample sizes typically used in diagnostic testing, and that post-test subjective questionnaires are tapping into factors beyond what happens during the test session.

53.4.6.2 Online Testing Measures

Online tests provide the potential for additional measures. Click stream data can show pages visited, page transitions, and how much time users spend on pages or key areas of pages (Albert, Tullis, and Tedesco 2010). For example, looking at pages visited during failed tasks can provide additional clues about design flaws. The larger sample sizes with online tests also make it possible to break the total population of participants into smaller segments, which is usually impossible with the small samples used in moderated tests.

53.4.7 New Ways of Reporting Test Results

In the early days of user testing, the test team almost always created a formal report and a highlight video tape. Testers needed those deliverables to communicate what they did, what they found, and to justify the testing method itself. Now, it is more common for the results to be communicated more informally, such as by scheduling a meeting soon after the last test session to discuss the results and/or creating a slide presentation for a briefing that may also contain sections of video from the sessions. Collaboration tools such as Wiki workspaces are also used to create “living” documentation to which subsequent design recommendations and user-interface concepts are added (Luef and Cunningham 2001).

53.5 TESTING STEPS OUT OF THE LABORATORY

With remote usability testing, the test administrator and participant are in different locations. Hartson and Castillo and colleagues began exploring remote usability testing as early as 1996 (Hartson et al. 1996; Castillo, Hartson, and Hix 1998). Tools to conduct remote usability testing were becoming available, and they saw the benefits of remote testing. Since then, technologies have improved and become less expensive, making it easier to conduct the tests. As a result, user experience professionals have continued to develop and explore methods of remote usability testing.

There are a number of advantages to remote testing:

- You can reach a worldwide population of participants because you are not limited to the local testing area. This may be especially helpful when there are not many users, and they are geographically dispersed.
- It is easier to get participants to volunteer because they do not have to travel.
- Participants work at their desks in their work environments, which may make them more comfortable and the testing more realistic. This can be especially helpful in recruiting disabled participants, who may find it difficult to travel or who use specific assistive technologies when they use the computer.
- You do not need a usability lab.

In the past, the technology to conduct such sessions was not good enough to allow usability specialists to get the information they need (Dumas 2003). That is no longer true because of several factors:

- The Internet has made it possible for usability specialists and participants to work together without installing special hardware or complex software on both the tester’s and the participant’s computers.
- There are tools available for instrumenting websites to collect usability measures automatically and to insert questions and ratings as the participants work.
- Collaboration software that works over the Internet makes it possible to share desktops and control the cursor.
- Recording software makes it possible to store good quality video and sound in files that are not large by today’s standards, often less than 50M for a 2-hour session.
- PC processors and RAM are fast enough to run both recording software and the application you are testing simultaneously. In addition, participants often have broadband or high-speed transmissions, so they are not limited by slow modem connections.

Remote testing takes two forms: (1) synchronous, in which the moderator and the participant work together, communicating over the phone or through their computer, and (2) asynchronous, in which the participants work on their own without the direct guidance of a moderator. Each has its strengths and weaknesses.

53.5.1 Synchronous Remote Testing

Synchronous remote testing is similar to a traditional usability laboratory test, except that the participant and tester are in different locations. The two methods tend to use similar protocols and similar methods of analysis. As a result, synchronous remote tests generally also involve the small numbers of participants.

With synchronous remote testing, the participant and moderator will use screen sharing software so that the moderator and other team members can observe what the participant is doing.

Typically, the administrator cannot see the participant (a webcam can be used, but usually is not). We do not yet know what the impact of not seeing the participant is, but one laboratory study indicates that usability specialists judge usability problems as less severe when they cannot see the participant's face (Lesaigle and Biers 2000).

Some remote testing configurations may present security problems. For example, participants could obtain screen shots without the knowledge of the moderator. In addition, allowing participants to share applications on computers inside your organization's firewall may be prohibited. Some organizations may be able to address this with a nondisclosure agreement, while others may require a special computer outside their firewalls.

53.5.2 Asynchronous Remote Testing

With asynchronous remote testing, participants complete the tasks on their own, and the test team reviews the session results later. Recently, the first book length discussion of this type of testing has appeared (Albert, Tullis, and Tedesco 2010). These are unmoderated tests. Asynchronous remote testing can be conducted by providing the participant with two browsers (one for the product or prototype and one for instructions). The instruction browser includes the tasks to be attempted, buttons to click at the beginning and end of a task, a free-form comment area, and questions or ratings to answer during or after each task. Asynchronous remote tests can also be conducted with tools specifically designed for that type of testing. Whichever arrangement is used, participants must be able to start and complete the entire test session on their own.

The primary advantage of asynchronous over synchronous testing is a larger sample size, because the number of participants is not limited by time requirements of the moderator. In addition, participants can complete the study at their convenience. For example, Tullis et al. (2002) tested 88 participants in a short period of time.

The disadvantage is that you cannot see or interact directly with the participants. However, in the Tullis et al. (2002) study, the participants provided an unexpectedly large volume of feedback in the free form comment field. These comments provided insight into the usability problems with the product.

53.5.3 Comparing Laboratory and Remote Testing

There have been just a few studies comparing results of usability tests conducted in a laboratory and remotely, and the results are not always consistent. Relating to performance measures, Tullis et al. (2002) reported no substantial difference between asynchronous testing and laboratory testing in terms of performance measures. However, West and Lehman (2006) found that asynchronous remote participants completed the tasks faster and were more likely to abandon a task than participants in a laboratory, but they showed similar success rates. Further, Thompson, Rozanski, and Haake (2004) also reported that asynchronous remote participants were faster. They also reported that these participants made fewer errors.

Regarding the number of problems identified, both Tullis et al. (asynchronous) and Thompson et al. (synchronous) found no difference with laboratory testing. However, in a study with blind participants, those in the asynchronous remote condition found fewer problems per website than those in the laboratory condition (each participant completed 2 tasks on each of 10 websites) (Petrie et al. 2006).

Clearly, we need to better understand the benefits and challenges of each method. As a result, research on this topic continues and is expanding into new domains such as remote testing with mobile devices.

53.5.4 Testing Mobile Devices

Conducting usability tests with mobile requires that testers be able to see both the screen of the device and the participants' hands. Early efforts used a computer-based emulator or a single camera pointed at a mobile device mounted on the table. These configurations captured the participants' interactions with the devices, but the experience was not realistic. Testers then developed creative solutions to capture the screen and the participants' hands. For example, both Catani (2003) and Schusteritsch, Wei, and LaRosa (2007) attached two small cameras to mobile devices, one to capture the screen (since many mobile devices have no "video out") and one to capture the participants' hands.

Another challenge is that mobile devices are intended to be used "on the go," not in the quiet office setting typically simulated in a usability test. Factors such as weather, signal strength, and background noise can all impact the users' experiences. In addition, mobile device users are often preoccupied by other tasks.

Several studies have evaluated the differences in the results from both laboratory and field usability tests, but there is little consistency in the findings. Kaikkaner et al. (2005) found exactly the same problems in both a laboratory and a field setting. Betiol and Cybis (2005) found more usability problems with a phone mounted to a desk than with a computer-based emulator or with a camera mounted on a mobile device used in the field. Duh, Tan, and Chen (2006) found more critical problems in the field than in the laboratory. On the other hand, Kjeldskov and Stage (2004) found more usability problems in the laboratory than in the field. However, the differences appear to be primarily in problems classified as "cosmetic," not in problems classified as "critical" or "serious." The great variety in the research results suggests that we need to continue to study this issue to better understand the methods of testing mobile devices.

53.6 ROLE OF THE TEST ADMINISTRATOR

Most usability specialists learn the skills of moderating tests through apprenticeship. They watch a few sessions, then moderate a few sessions under supervision. Quite quickly they move into a journeymen status during which they almost never receive feedback on their interaction skills unless they request it. In the first book published on the topic, Dumas and Loring (2008) have provided a systematic rationale for how to moderate a test session. They describe 10 rules for interacting with participants that put the first stake in the ground on the topic. The rules attempt to cover the common situations that moderators encounter rather than unusual incidents.

Dumas and Loring propose that moderators play three separate but overlapping roles:

1. The gracious host, who is responsible for making participants feel welcome from the moment they arrive to the moment they leave and who attends to their physical comfort, ensuring that the session goes smoothly and that they have a positive experience overall
2. The leader, who respects participants but who is clearly in charge of the direction and pacing of the session
3. The neutral observer, who is unbiased and objective and who keeps interactions to a minimum while providing support and encouragement to the participant when needed

Balancing those roles is one of the skills new moderators learn.

53.6.1 Training and Education of Moderators

Some of the early books on testing had chapters describing the skills needed and how to deal with selected situations. In the past 10 years, there have been a few Master's degree programs teaching moderating skills. But most usability professionals still learn on the job from more experienced moderators (Dumas 2007).

The usability profession has not established any educational or training qualification to become a moderator. Dumas and Loring (2008, p. 7) list the following qualifications:

- Understanding the basics of usability testing
- Interacting well with test participants (using our 10 rules)
- Ability to establish and maintain rapport with participants
- Lots of practice

Krug (2010) believes that all that is needed to be a competent moderator is a few hours of training in a workshop. He restricts his view to diagnostic testing. He says that he has never seen a bad moderator. He has challenged his readers to bring him a case in which a moderator has made a product less usable as a result of user testing. He believes that encouraging more moderators to run more tests is a path to making technology work better for its users. Clearly, we need more research on what makes a successful moderator.

53.7 FITTING TESTING INTO AN AGILE PROCESS

One of the important forces from outside of the user experience community that has had a major impact on its practices is Agile development (Frishberg 2010). Agile development methodology grew out the frustrations that the software industry has had managing the development process. After more than 25 years of trying, software was still released later than planned, over budget, and filled with bugs. Previous to Agile, the most common approach to development was the "waterfall" method, a sequential software development process in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, and maintenance. Starting about 2001, Agile development was a reaction against the waterfall model. The term "Agile" refers to a family of processes that share some common characteristics. Product requirements are addressed in a series of 2–4 week cycles by a dedicated team that is co-located. Each cycle ends with tested, working code. While code is documented, paper documents such as specifications are not part of the process.

While Agile development has begun to grow in popularity, it is unclear how traditional user experience methods, especially usability testing, can be integrated into it. Over the past few years, user experience professionals have been changing the way they work to remain players in these fast moving Agile cycles. Some of the important changes have been the following:

- Practicing iterative design and evaluation. The concept of iterative evaluation has been touted for decades, but the traditional waterfall model with traditional testing made iteration expensive and hard to justify (why are we testing again?). Iteration was, perhaps, the least practiced principle of user-centered design. Because iteration is at the foundation of the Agile model, user experience professionals have had to find a way to implement it. One approach is for the user experience team to be on a separate track from the coders, a track that is one cycle ahead (Lu, Rauch, and Miller 2010). While the user experience team is on Cycle 2, the coders are on Cycle 1. The user experience team does its user research and design concepts for Cycle 3 while conducting usability testing on the Cycle 1 user interface. The testing that is done is usually with very small samples and sometimes with internal staff rather than target users. Quantitative measures typically are not taken.

- Integration into the development team. The friction between user experience professionals and developers using the waterfall model kept user experience professionals on the outside looking in. Testing was often performed too late to impact design, and developers often viewed testers as people good at finding fault rather than fixing problems. The Agile method requires all members of the team to be co-located and to be engaged full-time. This face-to-face contact seems to create more cooperation and respect than was typical with the waterfall model.

Fitting testing into an Agile model was been facilitated by a new approach to testing (Medlock et al. 2005). Known as the RITE Method, it focuses on fixing designs rather than just finding problems. In outline, the method consists of the following:

- Key decision makers for the product participate in the study with the usability specialists.
- The team selects the tasks to be run and attends all sessions. As with traditional usability testing, users who are part of the target market for the product are recruited and sessions use the think aloud method.
- After each session, the usability specialist identifies problems and their severity. The team then decides whether they have enough data to verify each problem and how to refine the design to address the problem.
- The design team refines the design and tests it with the next participants.
- Problems are identified again, including whether the refinements have mitigated previous problems. If not, new refinements are created.
- The team decides which problems they can fix and which need to be examined in more detail or require resources that are not currently available.
- Additional participants are run until the major problems have been fixed or there are no more resources to continue.

With its emphasis on iteration and an integrated team, the RITE method fits nicely into the requirements of the Agile model (Douglass and Hylton 2010). Both Agile and RITE have the potential to change the way testing is performed and perceived. Those methods put pressure on testers to conduct tests quickly, to focus on fixing problems, and to require that developers be present during sessions.

As this chapter shows, since the early days of testing, there has been an emphasis on a faster process, scaled-down reporting, and getting modifications into the product. The traditional laboratory test with 5–8 participants, taking 2–4 weeks, with a report following some days later fit well into the waterfall model but not into the Agile model. The new approaches have some advantages in that they are more integrated into development and provide for iteration. But they also have the potential to make it convenient to test very small samples and to not use target users. It remains to be seen as testers move farther from the tradition testing basics whether diagnostic testing will remain as the most influential evaluation method. We desperately need some research to evaluate how effective testing is with these new models.

53.8 WEBSITE TESTING TOOLS

53.8.1 Eye Tracking

Eye tracking has slowly become more prevalent in usability testing. The technology has advanced to a point where it is noninvasive and almost unnoticeable to participants. Further, the software available to analyze the data also has improved. Although the equipment is still expensive, the prices are more affordable than in the past. Testers can even rent eye tracking equipment for short-term use at an even lower cost. These factors have led to an increase in the use of eye tracking in usability testing.

Eye trackers indicate where a participant is looking throughout a task or a whole test session. Eye trackers emit a pattern of infrared light (invisible to humans) and track the reflection of these patterns on the participants' eyes with special cameras. Participants no longer need to wear bulky head devices or stabilize their head when using an eye tracker. (Some eye trackers that can be used outside the laboratory are head-mounted but they are not as cumbersome as earlier models.)

Eye trackers can generate huge data files but vendors have developed sophisticated software that has greatly simplified the analysis process. This has been essential to the growing popularity of eye trackers, as they can sample data up to 120 times per second. Testers can now quickly determine the number or length of fixations on any particular area of a stimulus (such as a web page).

Testers can use eye tracking data in several ways. Eye trackers can be set up to allow observers to follow the participant's gaze during the test session. The test moderator can then tailor post-test debriefing questions based on patterns observed during the test. For example, if the participant spent a lot of time looking at a feature, the test moderator can ask what the participant thought of that feature.

Testers can also use the quantitative data generated by the eye tracker in post-test analyses. These eye tracking results can provide additional insights into participants' behaviors. The data can answer questions such as "Which areas of the page did participants look at most?" and "Were there areas they did not see it all?" For example, Albert and Tedesco (2010) used eye tracking measures to determine if self-reported awareness of items on a screen are reliable.

Running a usability test with eye tracking is not difficult but does require some additional planning. For example, testers will have to adjust their screening process slightly. Eye trackers may have difficulty tracking certain people, such as those who wear some styles of the bifocals. Also, the screeners should inform potential participants about the eye tracking and encourage them to bring whatever vision correction they need to see the screen easily. Participants who do not bring proper vision correction often sit too close or too far from the screen, where the equipment cannot track them.

In running a usability test with eye tracking, there are certain issues to consider:

- The informed consent should mention the eye tracking.
- The test protocol will have to include about 5–10 minutes at the beginning of each test session to calibrate the eye tracker to the participant.
- Scenarios for eye tracking tasks should not use a think aloud protocol. Participants look at the screen differently when they are thinking aloud (Bojko 2005). The tester may use eye tracking with some scenarios but not others to get a variety of information.

The analysis software for eye trackers can display the data in a variety of ways. Testers should understand the types of data they will be collecting and determine which are the most appropriate to address their issues (Bojko 2009; Poole and Ball 2005). Testers who want to use quantitative data should be sure to have enough participants to warrant statistical analyses (Goldberg and Wichansky 2002).

Eye tracking also has some disadvantages. It is difficult to conduct eye tracking studies with dynamic content, which includes not just video, but also objects such as cascading menus or pop-up message windows. The analysis software may present the results as if all the activity occurred on the original stimulus page.

Testing can be expensive, not just in terms of equipment, but also in terms of additional time to recruit participants, calibrate them during the test session, and analyze the results afterwards. In addition, because some participants cannot be tracked, the pool of possible participants becomes more limited. Testers may have to plan for additional participants in case some participants cannot be tracked (Schnipke and Todd 2000).

Despite these costs and challenges, eye tracking data can be very helpful in understanding participants' behavior. The data can help testers identify areas of confusion or point out objects participants missed entirely. Thus, although eye tracking is not standard usability laboratory equipment now, given the benefits of eye tracking, along with advancements in the technology, it is likely that the use of eye tracking will increase in the future.

53.8.2 FirstClick Testing

FirstClick usability testing is a method for evaluating the structure of a website. Wolfson et al. (2008) developed FirstClick testing as a way to conduct card sorting within the context of the actual website. They felt that the standard form of card sorting, using only labels and possibly brief descriptions for each "card," did not provide the same context as the website itself. They used it as a closed card sort, after designing wireframe options based on a more traditional open card sort.

In FirstClick testing, participants are given a task to complete. However, the scenario ends after they click on their first link. Researchers record the link selected and the time required in making a selection. Wolfson et al. also suggest having the participants rate their confidence after each selection. By aggregating data across participants, researchers can determine where users expect to start specific tasks. Researchers can see whether participants correctly selected the first link and whether the expectations were consistent.

To conduct a FirstClick test, researchers will need at least a somewhat functional wireframe of the homepage. The links must be active, but the second-level pages can just have a "task complete" message. With just a wireframe, researchers can conduct FirstClick testing fairly early in the development process, before the organization of the site has been established.

53.9 BASELINE AND COMPARISON TESTS

Some tests have a measurement focus, either for benchmarking a product's usability or comparing the usability of different products or versions. These performance-based tests tend to be summative and more like research experiments than a typical diagnostic test.

At present, the usability specialist's interpretation of summative usability test data plays a large role in evaluating the product's usability. Experienced usability professionals believe that they can make a relatively accurate and reliable assessment of a product's usability when considering the following:

- The product is stable.
- The number of participants is sufficiently large (larger than for most diagnostic tests).
- Participants are discouraged from making lengthy comments or evaluative statements in their think aloud protocol.
- The test administrator makes minimal interruptions to the flow of tasks.

The primary objective of a baseline test is to establish a standard against which other products or future versions of the product tested can be compared. By testing with the same set of tasks, a company can measure whether a new design has improved the usability of the product.

An important variation on the benchmark test is one focused primarily on comparing usability. Here the intention is to measure how usable a product is relative to some other product or to an earlier version of itself.

There are two variations:

- A diagnostic comparison test focused on finding as much as possible about a product's usability relative to a comparison product

- A summative comparison test intended to produce results that measure comparative usability and/or to find the winner

In both these tests, there are two important considerations:

- The test design must provide a valid comparison between the products.
- The selection of test participants, the tasks, and the way the test administrator interacts with participants must not favor any of the products.

As soon as the purpose of the test moves from diagnosis to comparison, the test design moves toward becoming more like a research experiment. In considering the design for the comparison, there are two important decisions:

- Will each participant use all the products, some of the products, or only one product?
- How many participants are enough to detect a statistically significant difference?

In the research methods literature, a design in which participants use all the products is called a "within-subjects" design, while in a "between-subjects" design, each participant uses only one product. If one uses a between-subjects design, one avoids having any contamination from product to product, but one needs to make sure that the groups who use each product are equivalent in important ways, and the sample size must increase. Because it is difficult to match groups on all the relevant variables, between-subject designs need to have enough participants in each group to wash out any minor differences. An important concern to beware of in the between-subjects design is the situation in which one of the participants in a group is especially good or bad at performing tasks; Gray and Salzman (1998) called this the "wildcard effect." If the group sizes are small, one superstar or dud could dramatically affect the comparison. With larger numbers of participants in a group, the wildcard has a smaller impact on the overall results. This phenomenon is one of the reasons that summative tests have larger sample sizes than diagnostic tests. The exact number of participants depends on the design and the variability in the data. Sample sizes in summative tests are closer to 20 in a group than the 5–8 that is common in diagnostic tests.

If one uses a within-subjects design in which each participant uses all the products, it eliminates the effect of groups not being equivalent and can have a smaller sample. However, one then has to worry about other problems, the most important being order and sequence effects and the length of the test session. (See Dumas (1998) for rules on counterbalancing.) One also has to be concerned about the test session becoming so long that participants get tired.

Perhaps the most important factor in the fairness of the comparison is the selection of tasks. The participants must perform the same tasks with the products. Anyone familiar with the products being compared is capable of selecting a sample of tasks that would favor one product. Consequently, some third party, perhaps an industry expert, who is not familiar with the details of the products but is familiar with the typical tasks users perform may be asked to select the tasks. Or a company conducting an internal comparison might ask a team independent of the test team to select the tasks.

The focus of the data analysis in a baseline or comparison task is usually on measures of performance and standardized subjective ratings rather than on qualitative measures that point to usability flaws.

53.10 TESTING WITH SPECIAL POPULATIONS

There is a growing literature about testing with special populations, including the following:

- International participants
- People with physical disabilities
- The elderly
- Children

This literature has been summarized by Dumas and Loring (2008). This section presents a brief summary of findings relevant to usability testing.

53.10.1 International Participants

Many manufacturers look for new customers across the globe. However, preparing a product for a new market may involve more than simply translating the language. Cultural differences can also impact appropriate design decisions such as color selections and the use of images. These differences can also impact the appropriate structure for web applications. Because of the significant differences across cultures, it is important to conduct usability testing with participants from all the target cultures.

International usability testing follows the principles and theories of generic usability testing. However, there are a variety of challenges with testing participants in other cultures that generally do not apply when testing in one's own culture. The challenges of communication and cultural differences are described below.

53.10.1.1 Communication

One of the most significant challenges with international usability tests is communication. Often, there are different languages. Sometimes the testers are bilingual, but often the tester must have helped recruiting participants, preparing test materials, conducting the test, analyzing the results, and writing the report. Nielsen (1996) and Vatrapu and Pérez-Quinones (2004) offer several suggestions including the following:

- Use employees of the company who live and work in that country. This may require training the employees to facilitate a usability test.

- Conduct the test in the participant's language using an interpreter.
- Hire a local usability firm.
- Run the test remotely.
- As a last resort, conduct the test yourself in your language, though this method is likely to be unnatural for the participant.

Tests that are conducted in the participant's language must be translated. Some testers prefer to have the translator work real-time during the test. The translator can either serve as a liaison between the tester and the participant (adding significant time to the test) or between the test administrator and participant (who are both speaking the same language) and the observers. The tester may also have to make arrangements to provide the test report in more than one language.

53.10.1.2 Cultural Differences

Other cultural differences may also impact a usability test. As noted earlier, Evers (2004) conducted think aloud tests and post-test interviews with a sample of 130 high school students from England, North America, the Netherlands, and Japan. There were several key differences including the finding that participants from Japan and the United Kingdom were uncomfortable thinking out loud. There may be gestures considered natural or friendly in one culture, but offensive in another. Vatrapu and Pérez-Quinones (2004) report that when both the participant and the test administrator were from the same culture, the participants engaged in more think aloud behavior and the usability tests revealed more problems.

53.10.2 Disabled Participants

Usability tests with disabled participants require careful planning. Testers must understand the participants' disabilities and adjust their procedures accordingly. Several researchers have published "lessons learned" from their experience with disabled participants (Coyne 2005; Grossnickle 2004; the Information Technology Technical Assistance and Training Center (ITTATC) 2004; Lepistö and Ovaska 2004; and Swierenga and Guy 2003). Some of these lessons include the following:

- Recruiting disabled participants is more time consuming than recruiting general population participants. Local organizations and support groups may be willing to help.
- Disabled participants may need assistance getting to the usability lab.
- Consent forms must be accessible to all participants.
- Blind participants may require electronic or Braille versions.
- Participants with learning or cognitive disabilities may require special consideration to ensure they understand the test and their rights.
- Deaf participants may require a sign language interpreter, who needs to be informed about the goals of the study.
- Participants with disabilities may require extra assistance understanding the tasks and may have trouble thinking aloud. Strain, Shaikh, and Boardman (2007) conducted concurrent think aloud tests with blind participants and found the audio from the screen reader interfered with the conversation.
- Participants with physical disabilities may require adaptive technology to interact with the computer. Be sure the devices are working before participants arrive.
- Because of the great variability in disabilities, it may take more participants than typical usability tests.
- It can be especially difficult to observe participants who use Braille readers, as there is currently no good way to follow what the participant is reading.

Overall, tests with disabled participants may take longer than expected; testers should schedule enough time so that participants are not rushed. Further, participation may be more taxing than for general population users, and so the test should limit the number of tasks evaluated (Coyne 2005). Finally, testers should ask participants before the test whether they need any special accommodations.

53.10.3 Elderly Participants

As the population ages, manufacturers are looking to expand their market to this growing population. Seniors are more active than ever. As a result, many manufacturers are working to ensure that their products are usable by their older users.

As people age, the diversity in their abilities increases. They may also have disabilities, such as those mentioned in the previous section. Many of the concerns and issues mentioned earlier also apply with elderly participants. In general, testers should be prepared for each participant, leaving plenty of time for each person.

There may also be generational issues. Testers should be aware of what their participants expect regarding social interaction and courtesy. Chisnell, Lee, and Redish (2005), Coyne (2005), and Tedesco, McNulty, and Tullis (2005) provide some guidance based on their experiences running usability test with older participants.

53.10.4 Children as Participants

When designing a product for children, usability tests must target children. Although the process is generally the same as with adult participants, there are a few important differences.

Recruiting children actually involves recruiting their parents. Patel and Paulsen (2002) suggest several good sources for recruiting. They recommend building rapport with organization leaders and parents. It is important to pay attention to the needs of both the parents and the child. Sometimes it is necessary to have the parents in the room during the test, especially for very young children. Investigators should be sure that the parents do not unnecessarily interfere with the test. However, investigators should be flexible, as each family will be different.

Investigators may want to alter the usability laboratory itself to be a better environment for children. Most usability laboratories use a standard office layout and décor. Although this is fine for testing adults, it is not the most welcoming to children. Making the room more "child friendly" can make children more comfortable and willing to participate.

The tasks should accommodate the abilities of children in the target age group. Investigators should consider (1) the wording of the instructions to be sure they are at an appropriate grade level and (2) whether the participants are old enough to complete the tasks. For example, children may not be able to perform a task and think aloud simultaneously. As mentioned earlier, Als, Jensen, and Skov (2005) used a technique with children called constructive interaction, in which children work in pairs on tasks. The pairs who used constructive interaction exposed more usability problems than the children who used thinking aloud.

Finally, what motivates adults does not always motivate children. Hanna, Risden, and Alexander (1997) suggest age-appropriate approaches for motivating young participants to continue. Most likely, the best approach for a preschooler is very different from that for a teenager.

Children can be unpredictable, so one or more members of the test team must understand the skills, abilities, and expectations of the children in the target user population. This will help testers to respond appropriately to unexpected situations.

53.11 HOW TESTS ARE ACTUALLY CONDUCTED

While there are many books and articles that describe how usability testing ought to be practiced, there have been few studies of how tests actually are conducted. The Comparative Usability Evaluations (CUE), especially the first two, inspected test reports from commercial usability laboratories (Molich et al. 1998; Molich et al. 2004). By reviewing the reports, the study authors saw the procedures used as well as the quality of the reports. There have been two other studies in which test sessions at commercial laboratories were observed and recorded (Boren and Ramey 2000; Norgaard and Hornbaek 2006). The results of these studies taken together are not encouraging. There is a large discrepancy between what testers actually do and what didactic texts say they should be doing.

The CUE studies looked at test reports from 13 organizations. No two reports were alike. They described tests from 4 to 50 participants with widely varying sets of tasks for the same product tested, leading or poorly designed task scenarios, different measures taken, and reports with few descriptions of the profiles of participants or procedures used.

Norgaard and Hornbaek watched and recorded 14 test sessions from seven different companies. They also recorded many discussions, analyses, and informal conversations among the usability evaluators before and after the sessions. They found that evaluators asked questions that were leading, questions asking participants to predict future outcomes, and questions that put words into the participants' mouths, such as "So … you feel more secure now … or?" (p. 215) There were two additional findings that are cause for concern. First, there was no systematic analysis while the results of a session were still fresh in evaluators' minds. Evaluators did not discuss findings during or directly after the sessions. Second, the behavior of evaluators indicated that they were confirming usability problems that they had found by inspecting the design themselves before the test started. Their tasks, questions, and probes were designed to support their own preconceived opinions about what the problems were. When participants' ratings disagreed with the evaluator's opinions, they were dismissed without further analysis.

While the Boren and Ramey and Noorgaard and Horbaek studies did not make other measures of participant's performance, a recent study has (Olmsted-Hawala et al. 2010). In that study, participants were assigned to various think-allowed conditions, ranging from "silent," with no think allowed or interaction with the test administrator, to "coaching," where the test administrator asked direct questions about the participant's thoughts and behaviors, which is what moderator's typically do in diagnostic testing. The results showed that when moderators are free to probe and ask questions, participants complete significantly more tasks and rate the product as more usable. That study is the first evidence that the moderator's behavior can change participants' behavior as well as the participants' perception of the product.

These studies suggest that usability testing as actually practiced is another important source of variability in usability measurement.

53.12 RELIABILITY OF USABILITY TESTING

Prior to about 1998, practitioners assumed that two equally competent teams conducting independent tests on the same product would have a large degree of overlap in the problems they detected, especially for problems judged to be severe. Jacobsen, Hertzum, and John (1998) were the first to study the reliability of testing in a laboratory experiment. They looked at how evaluators differ when analyzing the same usability test sessions. Four usability testers independently analyzed the same set of videotapes of four usability test sessions. Each session involved a user thinking aloud while solving tasks. Forty six percent of the problems were uniquely reported by single evaluators and all four evaluators agreed on only 20% of the problems. Furthermore, none of the top 10 most severe problems appeared on all four evaluators' lists.

In that same year, the first of the CUE studies appeared (Molich et al. 1998). It reported that of 141 unique problems found by four professional testing teams, only one problem appeared on all four lists. Subsequent CUE studies have also reported low levels of agreement (Molich et al. 2004; Molich and Dumas 2008).

Hertzum and Jacobsen (2001) conducted the first meta-analysis of reliability studies and termed the lack of agreement on problems "the evaluator effect." They also clarified the metric of agreement, recommending the any-2 agreement method. Any-2 agreement is the average of $|P_i \cap P_j| / |P_i \cup P_j|$ for all ½ $n(n - 1)$ pairs of evaluators—the total problems found in common divided by total problems found between two evaluators. Any-2 agreement has become the most commonly reported metric of reliability in assessments of usability evaluation. Using that metric, Hertzum and Jacobsen (2001) found only a 11% agreement among independent evaluators of a usability test.

To date, there have been more than two dozen papers with data on the reliability of testing, and they all show relatively low agreement rates. Several factors have been proposed to explain the low agreement:

- Evaluators use different tasks and task scenarios.
- Users explore different parts of the product.
- Participants are chosen based on different qualifications.
- Evaluators have different skills, experience, and training.
- Evaluators bring different biases to the test.
- There are no objective problem criteria.
- There is no metric for determining when two problems are the same or different.

53.12.1 Severe, Serious, or Just "Show Stoppers"

Several practitioners have proposed scheme for rating the severity of usability problems: Dumas and Redish (1999), Nielsen (1992), Rubin (1994), and Wilson and Coyne (2001). The schemes differ on a number of dimensions. In addition, many organizations have created their own scales. The reliability of severity scales has been questioned by several studies. Jacobsen, Hertzum, and John (1998) asked four experienced usability testers to watch tapes of the same usability test and then identify problems, including the Top 10 problems in terms of severity. None of the Top 10 severe problems appeared on all four evaluators' lists. Lesaigle and Biers (2000) reported a disappointing correlation coefficient (+0.16) among professional testers' ratings of the severity of the same usability problems in a usability test. Molich and Dumas (2008) found that 25% of the problems reported in common by two or more evaluation teams were classified into different severity categories.

The results of these studies strike a blow at one of the most often mentioned strengths of usability testing—its ability to uncover the most severe usability problems. At this point in time, we do not know whether the inconsistencies in severity judgments are the result of the poorly designed scales, the differing perceptions of usability specialists, the lack of training in how to make severity judgments, or all three.

53.12.2 Testing Is No Longer a Gold Standard

Several authors have proposed that usability testing be used as a gold standard against which to compare other evaluation methods (Andre, Williges and Hartson 2003; Sears 1997; Bailey, Allan, and Raiello 1992; Desurvire, Kondziela, and Atwood 1992). Their argument is that only problems identified by testing are true problems or hits. When other evaluation methods identify problems not found by testing, those problems are by that very fact not considered to be true problems. They are considered to be false positives, sometimes called false alarms. Those papers have been particularly harsh in their criticism of inspection by experts, such as heuristic evaluation.

There are two reasons to reject testing as a standard. First, as we have just described, independent tests find different subsets of problems. Second, Molich and Dumas (2008, p. 263) compared problem detection with testing and with expert inspection. They reported, "…there was no practical difference between the results obtained from usability testing and expert reviews for the issues identified. It was not possible to prove the existence of either missed problems or false alarms in expert reviews."

An issue not discussed in the literature comparing evaluation method is whether one should expect experts in usability evaluation to agree. There is a large body of literature on expertise showing that agreement among experts in most fields is low. It may be that disagreement among usability specialists is not any worse than it is among experts in medicine, biological, and social science disciplines (Shanteau 2001; Aboraya et al. 2006).

53.13 VALIDITY OF A USABILITY TEST

While there has been a good deal of research and analysis about the reliability of testing, there has been almost nothing written about its validity. Validity always has to do with whether a method does what it is supposed to do. There has never been a published study questioning whether testing finds problems. Perhaps the validity of usability testing has been ignored because, no matter how they are designed, tests always find strengths and weaknesses in a product. It has strong face validity. Testers believe when they have finished a test that they have uncovered the most important design flaws. But is finding problems enough?

To truly assess the validity of usability testing, we must first agree on what a usability test is supposed to do. Prior to the mid-1990s, the usability community used diagnostic tests primarily to uncover usability problems. The more problems found, the better and, of course, the tests should find the most severe ones. Because testing has never been viewed as the only usability evaluation method to apply during development and because, ideally, there are iterative tests performed, it was not essential or expected that one test would find all the problems.

The RITE method, discussed earlier, suggests two additional possibilities for goals:

1. A test should provide the data for and confirm the usability of an improved design.
2. A test should increase the commitment of a development team to user-centered design and its willingness to pursue it for future projects.

Fifteen years ago, Sawyer, Flanders, and Wixon (1996) proposed that the measure of validity for usability inspections should be how many of the problems that it identifies are actually fixed in the design. That criterion also could be applied to usability testing.

Until we sort out the importance of these goals (finding problems, creating an improved design, team building, and problems fixed in the design), we cannot fully understand the validity of what is arguably our most powerful usability assessment tool.

53.14 TESTING ETHICS

Informed consent is a method testers used to ensure that usability test participants have the information they need to decide whether to participate in the session. Millett, Friedman, and Felten (2001) state that “informed” requires the tester to be disclosing the necessary information in a manner that the participant can comprehend. They define “consent” to be the voluntary agreement to participate, made by someone competent to make such a decision.

Participants complete informed consent forms at the beginning of the test session. The forms themselves vary widely across organizations, but are generally expected to include the following information (Burmeister 2001):

- A brief description of what the participant will be expected to do
- A statement that participation is voluntary and that the participant can withdraw at any time without penalty
- Any potential risks the participant will be exposed to
- A description of any benefits either to the participant directly (including incentives) or to the population at large
- The name and contact information for the person responsible for the test
- How the test will handle all records from the test session (i.e., the extent to which data will be kept confidential), including the following:
    - Measures resulting from the test
    - Direct quotes from the participant
    - Video and/or audio recordings of the session (including whether the video will show the participant’s face)
    - Eye tracking data

Sometimes, testers use forms that allow participants to choose whether or not they will allow the testers to release video, quotes, and so on.

Completing the informed consent form usually only requires a participant to read and sign the form. Some testers follow the good practice of reviewing the form with the participants to be sure they understand and are aware of all the information.

However, in some cases, the informed consent process is not as straightforward.

Some disabilities make it difficult for participants to read, understand, and/or sign the form. When testing low-vision and blind users, the form should be presented in an accessible format. This may mean sending the form to participants ahead of time or providing Braille or large print versions (Henry 2007; Swierenga and Guy 2003). Testers may need to help physically disabled participants sign the form. In addition, testers may also need consent forms for sign language interpreters if they appear in any recordings (Henry 2007). When participants have cognitive disabilities, testers should be sure to provide the informed consent in a manner that each participant can understand.

When testing minors under the age of legal consent, the testers must get a signed consent form from a legal guardian, often a parent (Ellis, Quigley, and Power 2008). The guidelines from the U.S. Department of Health and Human Services (2008) allow guardians to fax in their forms. When the participants are old enough, it might be beneficial to have them sign a form as well, being sure to use age-appropriate language. This will help ensure that the minors understand their participation is voluntary.

When conducting one-on-one remote testing, it can be difficult to get signed consent forms before the test session starts. Dumas and Loring (2008) provide a sample electronic form that can be e-mailed to remote participants. For online testing, the testers usually do not know who the participants are and there is no audio or video recording. There still may be emotional or psychological risks to online participants, but that issue has not been explored in the literature.

When testing international populations, it is important to be sure the consent form is in a language each participant can understand. Also, there may be special requirements for information contained in the forms based on local regulations. As with disabled participants, you may also need consent forms for interpreters.

53.14.1 Additional Ethical Principles

The principles of informed consent and confidentiality that have been discussed in the HCI literature have been borrowed from ethical practices in biomedical research. We believe that, on the whole, testers have followed practices borrowed from biomedical research appropriately but have not been aware of some additional principles from social science research (House 1990).

Because of the often dramatic harm that biological and medical experimentation have caused in human history, ethical principles to protect participants have focused on costs and benefits that result from the application of procedures that occur during research studies in those areas. By analogy, the usability test has been treated as a variation on the research experiment. Consequently, the focus has been on informed consent being voluntary and knowledgeable and on confidentiality restricting the use of participants, name and, sometimes, video image. It has been assumed that risks of physical harm to participants in usability tests are minimal.

On the positive side, the sample informed consent forms in the literature describe the activities participants will be asked to perform, their right to withdraw at any time without penalty, the methods used for recording, and the restrictions on the use of data including the use of participants' names and images. But the possible risks of psychological or emotional harm and challenges to self-esteem are seldom mentioned. Perhaps, testers are afraid that mentioning those possibilities will bias participants to have a negative attitude toward the product being tested. The analogous situation in biomedical research would be not to mention a potentially harmful side effect because it might bias patients to expect such effects.

There is a large volume of literature that stresses the differences between biomedical and social science. There are at least two areas that are relevant to usability tests. First, in the social sciences, the researcher and the participant are often presented as equal partners in the investigation. They work together as colleagues (Murphy et al. 1998). This is the way diagnostic usability testing typically is framed. For example, in formative tests, the test administrator is more engaging and active toward participants. While this approach is intended to make participants feel empowered and more comfortable, it can do just the opposite when participants struggle and fail at tasks. When that happens, it presents the tester and participants with a situation that is not covered by the typical informed consent statement. The hidden assumption behind letting participants fail is that, in a utilitarian accounting of harm, it is better that a few participants fail so that potentially many future users will not fail (see Dumas and Loring 2008). This utilitarian approach to ethics runs counter to a different approach that says that it is unethical to use a harmful means to achieve a beneficial end (Macklin 1982). According to that approach, knowingly causing distress and possibility lowering self-esteem cannot be justified without informed consent. At a minimum, informed consent forms for usability testing should describe the possibility of emotional distress.

A second difference is that violations to participants' confidentiality may come during the reporting phase, which may occur long after the test sessions (Hammersley and Atkinson 1995). While test reports almost never mention participants by name, their user role is often described. A common strategy for emphasizing the priority of a usability problem is to quote participants' negative descriptions of the product or even the company developing it. In tests with small populations that are performed on internal rather than commercial products such quotes may be attributable to particular individuals who then face the embarrassment of exposure. In addition to quotes, it is now technically easy to attach a segment of tape to a slide presentation showing the quote or task failures. In these situations, participants are used as ammunition in the battle between testers and developers over whether changes will be made to the product. Testers need to be aware of the ethics of these situations and take extra precautions to ensure that the identity of participants is not revealed.

53.15 CONCLUSION

Usability testing has evolved in line with changes in the user experience field. For example, practitioners have been exploring ways to work faster and cheaper and to be less formal in their preparation and reporting. In addition, we are just beginning to understand the impact of long-accepted think aloud methods. We now have a better understanding of the standard usability measures, but we also have new technologies, such as eye tracking, which provide new sources of data. Some issues, such as the number and types of participants to use, continue to be debated with no clear resolution. As evidence of this, there has been a push to find ways to conduct tests with both local, convenient participants (e.g., hallway testing) and diverse participants (remote testing). So although there has been progress on many fronts, there are still many areas left to explore.

REFERENCES

Aboraya, A., E. Rankin, C. France, A. El-Missiry, and C. John. 2006. The reliability of psychiatric diagnosis revisited: The clinician's guide to improve the reliability of psychiatric diagnosis. *Psychiatry* 3(1):41–50.

Albert, W., and D. Tedesco. 2010. Reliability of self-reported awareness measures based on eye tracking. *J Usability Stud* 5(2):50–64.

Albert, W., T. Tullis, and D. Tedesco. 2010. *Beyond the Usability Lab: Conducting Large-Scale Online User Experience Studies*. Burlington, MA: Morgan Kaufmann Publishers.

Als, B., J. Jensen, and M. Skov. 2005. Comparison of think-aloud and constructive interaction in usability testing with children. In *Proceedings of the Conference on Interaction Design and Children*, 9–16. (Boulder, CO), New York: The Association for Computing Machinery.

Andre, T., R. Williges, and H. Hartson. 2003. The effectiveness of usability evaluation methods: Determining the appropriate criteria. In *Proceedings of the Human Factors and Ergonomics Society, 43rd Annual Meeting*, 1090–4. (Denver, CO), Santa Monica, CA: The Human Factors and Ergonomics Society.

Bailey, R. W., R. W. Allan, and P. Raiello. 1992. Usability testing vs. heuristic evaluation: A head-to-head comparison. In *Proceedings of the Human Factors Society, 36th Annual Meeting*, 409–13. (Atlanta, GA), Santa Monica, CA: The Human Factors Society.

Betiol, A. H., and W. Cybis. 2005. Usability testing of mobile devices: A comparison of three approaches. In *Proceedings of the Tenth IFIP TC13 International Conference on Human-Computer Interaction*, 470–81. (Rome, Italy), The IFIP Technical Committee on Human-Computer Interaction.

Birru, M. S., V. M. Monaco, L. Charles, H. Drew, V. Njie, T. Bierria, E. Detlefsen, and R. A. Steinman. 2004. Internet usage by low-literacy adults seeking health information: An observational analysis. *J Med Internet Res* 6(3):e25. www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1550604 (accessed November 29, 2011).

Bojko, A. 2005. Eye tracking in user experience testing: How to make the most of it. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–9. (Montreal, Canada), Bloomingdale, IL: The Usability Professionals' Association.

Bojko, A. 2009. Informative or misleading? Heatmaps deconstructed. In *Human-Computer Interaction*, ed. J. Jacko, 30–9. Heidelberg, Germany: Springer-Verlag.

Boren, M., and J. Ramey. 2000. Thinking aloud: Reconciling theory and practice. *IEEE Trans Prof Commun* 43(3):261–78.

Branaghan, R. 1997. Ten tips for selecting usability test participants. *Common Ground* 7:3–6.

Burmeister, O. K. 2001. Usability testing: Revisiting informed consent procedures for testing Internet sites. In *Second Australian Institute Conference on Computer Ethics*, 3–9. (Sydney, Australia), Darlinghurst, Australia: The Australian Computer Society, Inc.

Castillo, J. C., H. R. Hartson, and D. Hix. 1998. Remote usability evaluation: Can users report their own critical incidents? In *Proceedings of Human Factors in Computing Systems*, 253–354. (Los Angeles, CA), New York: The Association for Computing Machinery.

Catani, M. B. 2003. Observation methodologies for usability tests of handheld devices. In *Proceedings of The Usability Professionals' Association Annual Meeting*, 1–6. (Scottsdale, AZ), Bloomingdale, IL: The Usability Professionals' Association.

Chisnell, D., A. Lee, and J. Redish. 2005. *Recruiting and Working with Older Participants*, *American Association of Retired Persons*. www.aarp.org/olderwiserwired/oww-features/Articles/a2004-03-03-recruiting-participants.html (accessed October 13, 2005).

Coyne, K. P. 2005. Conducting simple usability studies with users with disabilities. In *Proceedings of HCI International*, 890–3. (Las Vegas, NV), Mahwah, NJ: Lawrence Erlbaum Associates.

Desurvire, H. W., J. M. Kondziela, and M. E. Atwood. 1992. What is gained and lost when using evaluation methods other than empirical testing. In *People and Computers VII*, ed. A. Monk, D. Diaper, and M. D. Harrison, 89–102. Cambridge, MA: Cambridge University Press.

Douglass, R., and K. Hylton. 2010. Get it RITE. *User Exp* 9:12–3.

Duh, H. B.-L., G. C. B. Tan, and V. H. Chen. 2006. Usability evaluation for mobile device: A comparison of laboratory and field tests. In *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services*, 181–6. (Helsinki, Finland), New York: The Association for Computing Machinery.

Dumas, J. 1998. Usability testing methods: Using test participants as their own controls. *Common Ground* 8:3–5.

Dumas, J. 2001. Usability testing methods: Think aloud protocols. In *Design by People for People: Essays on Usabilit*y, ed. R. Branaghan, 119–30. Chicago, IL: Usability Professionals' Association.

Dumas, J. 2003. Usability evaluation from your desktop. *Assn Inf Syst (AIS) SIGCHI Newsletter* 2(2):7–8.

Dumas, J. 2007. The great leap forward: The birth of the usability profession (1988–1993). *J Usability Stud* 2(2):54–60.

Dumas, J., and J. Fox. 2007. Usability testing: Current practice and future directions. In *The Human-Computer Interaction Handbook*, ed. J. Jacko and A. Sears 2nd ed., 1129–49. Mahwah, NJ: Lawrence Erlbaum, Associates.

Dumas, J., and B. Loring. 2008. *Moderating Usability Tests: Principles and Practices for Interacting*. San Francisco, CA: Morgan Kaufman.

Dumas, J., and G. Redish. 1993. *A Practical Guide to Usability Testing (1st ed.).* London: Intellect Books.

Dumas, J., and G. Redish. 1999. *A Practical Guide to Usability Testing* (Rev. ed.). London: Intellect Books.

Ebling, M., and B. John. 2000. On the contributions of different empirical data in usability testing. In *Proceedings of Designing Interactive Systems*, 289–96. (Brooklyn, NY), New York: The Association for Computing Machinery.

Eger, N., L. J. Ball, R. Stevens, and J. Dodd. 2007. Cueing retrospective verbal reports in usability testing through eye-movement replay. In *People and Computers XXI—HCI … but not as we know it: Proceedings of HCI 2007*, ed. L. J. Ball, M. A. Sasse, C. Sas, T. C. Ormerod, A. Dix, P. Bagnall, and T. McEwan. Swindon: The British Computer Society.

Ellis, K., M. Quigley, and M. Power. 2008. Experiences in ethical usability testing with children. *J Inf Technol Res* 1(3):1–13.

Ericsson, K. A., and H. A. Simon. 1993. *Protocol Analysis: Verbal Reports as Data*. Cambridge, MA: MIT Press.

Evers, V. 2004. Cross-cultural applicability of user evaluation methods. A case study amongst Japanese, North-American, English and Dutch users. In *Proceedings of Human Factors in Computing Systems*, 740–1. (New Orleans, LA), New York: The Association for Computing Machinery.

Faulkner, L. 2003. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behav Res Methods Instrum Comput* 35(3):379–83.

Frishberg, N. 2010. Agile and UX. *User Exp* 9:4.

Frøkjær, E., M. Hertzum, and K. Hornbæk. 2000. Measuring usability: Are effectiveness, efficiency, and satisfaction really correlated? In *Proceedings of Human Factors in Computing Systems*, 345–52. (Fort Lauderdale, FL), New York: The Association for Computing Machinery.

Frøkjær, E., and K. Hornbæk. 2005. Cooperative usability testing: Complementing usability tests with user-supported interpretation sessions. In *Proceedings of Human Factors in Computing Systems*, 1383–6. (Denver, CO), New York: The Association for Computing Machinery.

Goldberg, J. H., and A. M. Wichansky. 2002. Eye tracking in usability evaluation: A practitioner's guide. In *The Mind's Eyes: Cognitive and Applied Aspects of Eye Movements*, ed. J. Hyönä, R. Radach, and H. Deubel, 493–516. Oxford: Elsevier Science.

Gray, W., and M. Salzman. 1998. Damaged merchandise? A review of experiments that compare usability methods. *Hum Comput Interact* 13:203–335.

Grossnickle, M. M. 2004. How many users with disabilities should you include when conducting a usability test for accessibility? Idea Market presented at *The Usability Professionals' Association Annual Meeting* www.upassoc.org/usability_resources/conference/2004/im_martinson.html (accessed September 13, 2005).

Hammersley, M., and P. Atkinson. 1995. *Ethnography: Principles in Practice*. London: Routledge.

Hancock, P., A. Pepe, and L. Murphy. 2005. Hedonomics: The power of positive and pleasurable ergonomics. *Ergon Des* 13(1):8–14.

Hanna, L., K. Risden, and K. J. Alexander. 1997. Guidelines for usability testing with children. *Interactions* 4:9–14.

Hartson, H. R., J. C. Castillo, J. Kelso, and W. Neale. 1996. Remote evaluation: The network as an extension of the usability laboratory. In *Proceedings of Human Factors in Computing Systems*, 228–35. (Vancouver, Canada), New York: The Association for Computing Machinery.

Henry, S. L. 2007. Just ask. www.uiaccess.com/accessucd/index.html (accessed February 12, 2010).

Hertzum, M., and N. E. Jacobsen. 2001. The evaluator effect: A chilling fact about usability evaluation methods. *Int J Hum Comput Interact* 13(4):421–43.

Hornbæk, K., and E. Law. 2007. Meta-analysis of correlations among usability measures. In *Proceedings of Human Factors in Computing Systems*, 617–26. (San Jose, CA), New York: The Association for Computing Machinery.

House, E. 1990. An ethics of qualitative field studies. In *The Paradigm Dialog*, ed. E. Gaba, 158–201. Newbury Park, CA: Sage.

Information Technology Technical Assistance and Training Center (ITTATC), Georgia Institute of Technology. 2004. *Planning Usability Testing for Accessibility*. www.ittatc.org/technical/access-ucd/ut_plan.php (accessed September 13, 2005).

Jacobsen, N., M. Hertzum, and B. E. John. 1998. The evaluator effect in usability studies: Problem detection and severity judgments. In *Proceedings of the Human Factors and Ergonomics Society, 42nd Annual Meeting*, 1336–40. (Chicago, IL), Santa Monica, CA: The Human Factors and Ergonomics Society.

Jordan, P. 2002. The personalities of products. In *Pleasure with Products*, ed. W. Green and P. Jordan, 19–48. London: Taylor & Francis.

Kaikkaner, A., A. Kekalainen, M. Canker, T. Kalliot, and A. Kankainen. 2005. Usability testing of mobile applications: A comparison between laboratory and field studies. *J Usability Stud* 1:4–16.

Karat, J. 2003. Beyond task completion: Evaluation of affective components of use. In *The Human-Computer Interaction Handbook*, ed. J. Jacko and A. Sears, 1152–64. Mahwah, NJ: Lawrence Erlbaum, Assoc.

Kjeldskov, J., and J. Stage. 2004. New techniques for usability evaluation of mobile systems. *Int J Hum Comput Stud* 60(5–6):599–620.

Krahmer, E., and N. Ummelen. 2004. Thinking about thinking aloud: A comparison of two verbal protocols for usability testing. *IEEE Trans Prof Commun* 47(2):105–17.

Krug, S. 2010. *Rocket Surgery Made Easy*. Berkeley, CA: New Riders.

Law, C., and G. Vanderheiden. 2000. Reducing sample sizes when user testing with people who have and who are simulating disabilities: Experiences with blindness and public information kiosks. In *Proceedings of the IEA 2000/HFES 2000 Congress*, 26, 157–60. (San Diego, CA), Santa Monica, CA: The Human Factors and Ergonomics Society.

Lepistö, A., and S. Ovaska. 2004. Usability evaluation involving participants with cognitive disabilities. In *Proceedings of NordiCHI*, 305–8. (Tempere, Finland), New York: The Association for Computing Machinery.

Lesaigle, E. M., and D. W. Biers. 2000. Effect of type of information on real-time usability evaluation: Implications for remote usability testing. In *Proceedings of the IEA 2000/HFES 2000 Congress*, 37, 585–8. (San Diego, CA), Santa Monica, CA: The Human Factors and Ergonomics Society.

Lewis, J. 1994. Sample size for usability studies: Additional considerations. *Hum Fact* 36:368–78.

Lewis, J. 2001. Evaluation of procedures of adjusting problem discovery rates estimates from small samples. *Int J Hum Comput Interact* 71(1):57–78.

Lindgaard, G., and J. Chattratichart. 2007. Usability testing: What have we overlooked? In *Proceedings of Human Factors in Computing Systems*, 1415–24. (San Jose, CA), New York: The Association for Computing Machinery.

Lu, C., T. Rauch, and L. Miller. 2010. Agile teams: Best practices for agile development. *User Exp* 9:6–10.

Luef, B., and W. Cunningham. 2001. *The Wiki Way: Quick Collaboration on the Web.* Reading, MA: Addison-Wesley, Inc.

Macklin, R. 1982. The problem of adequate disclosure in social science research. In *Ethical Issues in Social Science Research*, ed. T. Beauchamp, R. Faden, R. Wallace, and L. Walters, 193–214. Baltimore, MD: Johns Hopkins.

Medlock, M., D. Wixon, M. McGee, and D. Welsh. 2005. The rapid iterative test and evaluation method: Better products in less time. In *Cost-Justifying Usability: An Update for the Information Age*, 489–517. New York: Morgan Kaufman Publishers.

Millett, L. I., B. Friedman, and E. Felten. 2001. Cookies and web browser design: Toward realizing informed consent online. In *Proceedings of Human Factors in Computing Systems*, 46–52. (Seattle, WA), New York: The Association for Computing Machinery.

Molich, R., N. Bevan, I. Curson, S. Butler, E. Kindlund, D. Miller, and J. Kirakowski. 1998. Comparative evaluation of usability tests. In *Proceedings of the Usability Professionals' Association Annual Meeting.* Bloomingdale, IL: The Usability Professionals' Association.

Molich, R., and J. Dumas. 2008. Comparative usability evaluation (CUE-4). *Behav Inf Technol* 27(3):263–81.

Molich, R., R. Meghan, K. Ede, and B. Karyukin. 2004. Comparative usability evaluation. *Behav Inf Technol* 23:65–74.

Murphy, E., R. Dingwall, D. Greatbatch, S. Parker, and P. Watson. 1998. Qualitative research methods in health technology assessment: A review of the literature. *Health Technol Assess* 2(16):1–272.

Murphy, L., K. Stanney, and P. Hancock. 2003. The effect of affect: The hedonic evaluation of human-computer interaction. In *Proceedings of the Human Factors and Ergonimics Society 47th Annual Meeting*. 764–7. Santa Monica, CA: The Human Factors and Ergonomics Society.

Nielsen, J. 1992. Finding usability problems through heuristic evaluation. In *Proceedings of Human Factors in Computing Systems*, 373–80. (Monterey, CA), New York: The Association for Computing Machinery.

Nielsen, J. 1996. *International Usability Testing.* www.useit.com/papers/international_usetest.html (accessed September 13, 2005).

Nielsen. 2000. *Why you Only Need to Test with 5 Users*. www.useit.com/alertbox/20000319.html (accessed February 22, 2009).

Nielsen, J., T. Clemmensen, and C. Yssing. 2002. Getting access to what goes on in people's heads: Reflections on the think-aloud technique. In *Proceedings of NordiCHI*, 101–10. (Aarhus, Denmark), New York: The Association for Computing Machinery.

Norgaard, M., and K. Hornbaek. 2006. What do usability evaluators do in practice? An explorative study of think-aloud testing. In *Proceedings of Designing Interactive Systems*, 209–18. (University Park, PA), New York: The Association for Computing Machinery.

Olmsted-Hawala, E., S. Hawala, E. Murphy, and K. Ashenfelter. 2010. Think-aloud protocols: A comparison of three think-aloud protocols for use in testing data-dissemination web sites for usability. In *Proceedings of Human Factors in Computing Systems*, 2381–90. (Atlanta, GA), New York: The Association for Computing Machinery.

Pagulayan, R., K. Keeker, D. Wixon, R. Romero, and T. Fuller. 2003. User-centered design in games. In *The Human-Computer Interaction Handbook*, ed. J. Jacko and A. Sears, 883–906. Mahwah, NJ: Lawrence Erlbaum, Assoc.

Patel, M., and C. A. Paulsen. 2002. Strategies for recruiting children for usability tests. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–4. (Orlando, FL), Bloomingdale, IL: The Usability Professionals' Association.

Petrie, H., F. Hamilton, N. King, and P. Pavan. 2006. Remote usability evaluations with disabled people. In *Proceedings of Human Factors in Computing Systems*, 1133–41. (Montreal, Canada), New York: The Association for Computing Machinery.

Poole, A., and L. J. Ball. 2005. Eye tracking in human-computer interaction and usability research: Current status and future prospects. In *Encyclopedia of a Human-Computer Interaction*, ed. C. Ghaoui, 211–19. Hershey, PA: Idea Group.

Pruitt, J., and T. Adlin. 2005. *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*. San Francisco, CA: Morgan Kaufman.

Quesenbery, W. 2004. "Balancing the 5Es: Usability." *Cutter IT J* 17(2):4–11.

Quesenbery, W. 2005. The five dimensions of usability. In *Content and Complexity: Information Design in Technical Communication*, ed. M. Albers, B. Mazur, 81–102. Mahwah, NJ: Lawrence Erlbaum & Associates.

Redish, J. 2007. Expanding usability testing to evaluate complex systems. *J Usability Stud* 2:102–11.

Redish, J. C., and J. Scholtz. 2007. Evaluating complex information systems for domain experts. Paper presented at *HCI and Information Design to Communicate Complex Information*, 1–23. Memphis, TN: University of Memphis.

Rosenbaum, S., J. Rohn, and J. Humburg. 2000. A toolkit for strategic usability: Results from workshops, panels, and surveys. In *Proceedings of Human Factors in Computing Systems*, 337–44. (The Hague, Netherlands), New York: The Association for Computing Machinery.

Rubin, J. 1994. *Handbook of Usability Testing.* New York: John Wiley & Sons, Inc.

Sauro, J., and J. Dumas. 2009. Comparison of three one-question, post-task usability questionnaires. In *Proceedings of Human Factors in Computing Systems*, 1599–608. (Boston, MA), New York: The Association for Computing Machinery.

Sauro, J., and J. Lewis. 2009. Correlations among prototypical usability metrics: Evidence for the construct of usability. In *Proceedings of Human Factors in Computing Systems*, 1609–18. (Boston, MA), New York: The Association for Computing Machinery.

Sawyer, P., A. Flanders, and D. Wixon. 1996. Making a difference—the impact of inspection. In *Proceedings of Human Factors in Computing Systems*, 378–82. (Vancouver, British Columbia, Canada), New York: The Association for Computing Machinery.

Schnipke, S. K., and M. W. Todd. 2000. Trials and tribulations of using an eye-tracking system. In *Proceedings of Human Factors in Computing Systems*, 185–6. (The Hague, Netherlands), New York: The Association for Computing Machinery.

Schusteritsch, R., C. Y. Wei, and M. LaRosa. 2007. Towards the perfect infrastructure for usability testing on mobile devices. In *Proceedings of Human Factors in Computing Systems*, 1839–44. (San Jose, CA), New York: The Association for Computing Machinery.

Shanteau, J. 2001. What does it mean when experts disagree? In *Linking Expertise and Naturalistic Decision Making*, ed. E. Salas and G. Klein, 229–44. Mahwah, NJ: Lawrence Erlbaum Associates.

Sears, A. 1997. Heuristic walkthroughs: Finding the problems without the noise. *Int J Hum Comput Interact* 9:213–34.

Spolsky, J. 2000. *The Joel Test: 12 Steps to Better Code*. www.joelonsoftware.com/articles/fog0000000043.html (accessed November 29, 2011).

Strain, P., A. D. Shaikh, and R. Boardman. 2007. Thinking but not seeing: Think-aloud for non-sighted users. In *Proceedings of Human Factors in Computing Systems*, 1851–6. (San Jose, CA), New York: The Association for Computing Machinery.

Swierenga, S. J., and T. Guy. 2003. Session logistics for usability testing of users with disabilities. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–6. (Scottsdale, AZ), Bloomingdale, IL: The Usability Professionals' Association.

Teague, R., and H. Whitney. 2002. What's love got to do with it? *User Exp* 1:6–13.

Tedesco, D., M. McNulty, and T. Tullis. 2005. Usability testing with older adults. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–8. (Montreal, Canada), Bloomingdale, IL: The Usability Professionals' Association.

Tedesco, D., and T. Tullis. 2006. A Comparison of methods for eliciting post-task subjective ratings in usability testing. In *Proceedings of the Usability Professionals Association Annual Meeting*, 1–9. (Broomfield, Colorado), Bloomingdale, IL: The Usability Professionals' Association.

Thompson, K., E. Rozanski, and A. Haake. 2004. Here, there, anywhere: Remote usability testing that works. In *Proceedings of SIGITE*, 132–7. (Salt Lake City, UT), New York: The Association for Computing Machinery.

Tullis, T., S. Flieschman, M. McNulty, C. Cianchette, and M. Bergel. 2002. An empirical comparison of lab and remote usability testing of web sites. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–5. (Orlando, FL), Bloomingdale, IL: The Usability Professionals' Association.

Tullis, T., and J. Stetson. 2004. A comparison of questionnaires for assessing website usability. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–12. (Minneapolis, MN), Bloomingdale, IL: The Usability Professionals' Association.

Turner, C., J. R. Lewis, and J. Nielsen. 2006. Determining usability test sample size. In *International Encyclopedia of Ergonomics and Human Factors*, ed. W. Karwowski, 3084–8. Boca Raton, FL: CRC Press.

U.S. Department of Health and Human Services. 2008. *Office for Human Research Protections (OHRP): OHRP Informed Consent Frequently Asked Questions*. www.answers.hhs.gov/ohrp/categories/1566 (accessed February 25, 2010).

van den Haak, M. J., M. D. T. de Jong, and P. J. Schellens. 2003. Retrospective vs. concurrent think aloud protocols: Testing the usability of an online library catalogue. *Behav Inf Technol* 22(5):339–51.

Vatrapu, R., and M. A. Pérez-Quiñones. 2004. *Culture and International Usability Testing: The Effects of Culture in Structured Interviews.* Technical Report cs.HC/0405045. Computing Research Repository (CoRR). http://arxiv.org/pdf/cs/0405045v1 (accessed October 4, 2005).

Virzi, R. A. 1990. Streamlining the design process: Running fewer subjects. In *Proceedings of the Human Factors Society, 34th Annual Meeting*, 291–4. (Orlando, FL), Santa Monica, CA: The Human Factors and Ergonomics Society.

Virzi, R. A. 1992. Refining the test phase of usability evaluation: How many subjects is enough? *Hum Fact* 34:457–68.

Virzi, R. A., J. F. Sorce, and L. B. Herbert. 1993. A comparison of three usability evaluation methods: Heuristic, think-aloud, and performance testing. In *Proceedings of the Human Factors and Ergonomics Society, 37th Annual Meeting*, 309–13. (Seattle, WA), Santa Monica, CA: The Human Factors and Ergonomics Society.

West, R., and K. R. Lehman. 2006. Automated summative usability studies: An empirical evaluation. In *Proceedings of Human Factors in Computing Systems*, 631–9. (Montreal, Canada), New York: The Association for Computing Machinery.

Wilson, C. E., and K. P. Coyne. 2001. Tracking usability issues: To bug or not to bug? *Interactions* 8:15–9.

Wolfson, C. A., R. W. Bailey, J. Nall, and S. Koyani. 2008. Contextual card sorting (or FirstClick testing): A new methodology for validating information architectures. In *Proceedings of the Usability Professionals' Association Annual Meeting*, 1–6. (Baltimore, MD), Bloomingdale, IL: The Usability Professionals' Association.

54 Usability for Engaged Users

The Naturalistic Approach to Evaluation

David Siegel

CONTENTS

54.1 INTRODUCTION

This chapter is intended to give the reader an introduction to naturalistic usability, sometimes called field usability. The term "naturalistic usability" includes a range of approaches to evaluation that attempt to provide a more holistic and realistic assessment of usability than is possible in a conventional usability laboratory study. It attempts to come as close as possible to measuring usability as people actually experience it by exploring the user's interaction with technology in the user's natural context, based on the user's own goals and materials. These approaches also allow you to study both usability and utility conjointly and to explore both initial usability (e.g., discovery) and continuing task performance in a more balanced way than is usually possible in a laboratory study (Dray and Siegel 2009).

It will help focus the discussion to start by contrasting naturalistic usability with traditional laboratory-based usability testing on the one hand, and with two other approaches to user experience research in the field (ethnography and contextual inquiry) on the other hand. No single approach can provide more than a slice of the truth, and so these contrasts are not meant to show that naturalistic usability is better than other approaches, but rather to clarify how it complements them. After contrasting these approaches, we turn to discussion of some of the key difficulties that any attempt to assess usability naturalistically entails and describe strategies to address them. Next, we examine some case examples of naturalistic evaluation projects to show how different strategies might work in different circumstances. Finally, we consider how and when naturalistic evaluation fits into the product development lifecycle.

54.2 METHODOLOGICAL CONTRASTS

Arguments about nomenclature for professional communities and the practices that define them are extremely common. As professional communities grow, they tend to realize that their focus is more broadly relevant and to expand their mandate. Thus, they cross into each other's territory. At the same time, communities feel a need to defend their distinctiveness. This process has certainly occurred within the field of user experience (Siegel and Dray 2005). This can lead to endless turf battles, co-option of things that have to be done under other rubrics, and games of logical one-upsmanship (i.e., "My method subsumes your method.") Certainly, people may debate whether what follows draws the lines correctly. Please keep in mind that experienced and inventive practitioners can do a wide range of things beyond traditional methodological boundaries. Be aware, too, that the names of the methods do not map perfectly to the disciplinary titles of practitioners, that textbook definitions do not always agree, and that practitioners do not necessarily follow the textbooks. People who describe themselves as ethnographers can spend some time doing what looks very much like contextual inquiry, just as people doing usability evaluation in the laboratory may get insights about the deep goals of their users that are more traditionally thought of as the province of ethnography. Similarly, people doing naturalistic usability in the field may at times look like they are doing ethnography or contextual inquiry or even things not that different from standard laboratory evaluations. The discussion of methodological contrasts that follow is intended to emphasize contrasts for didactic purposes and to help the reader think about dimensions of difference. A given example of a real research project may be difficult to classify, and which zone in the space of methodologies belongs to whom is not the most important concern here.

54.2.1 Conventional Laboratory Usability Evaluation

Much, even most, usability testing takes place in a laboratory, which is a simulated environment, using simulated tasks. Typically, these studies are carried out using a sample of people who are not actual users, but who are instead selected to resemble real users on dimensions that the researcher thinks are relevant, including things such as the background and knowledge they bring to the task. This approach makes perfect sense if the goal is to come up with generalizable findings, like inferences about usability for people interacting with the technology for a wide range of tasks in a wide range of circumstances. An emphasis on generalizability requires constructing representative tasks in a form that is generic enough for the following:

- A sample of users drawn from different real contexts to understand and engage with them
- The findings to have implications across a class of similar real situations, despite presumably superficial differences

Paradoxically, although the goal is generalizability to the real world, the tradeoffs involved in creating broad representativeness in laboratory usability evaluation can actually *limit* its applicability to the real world. Doing the evaluation in a simulation environment means there are more layers of inference between observed behavior or problems in the test situation and the real-life contexts where usability problems really matter (Hartson, Andre, and Williges 2003). The following characteristics of laboratory usability evaluation (whether formative or summative) are some of the ways it tends to systematically exclude many factors that influence how usability is experienced in the real world:

- Lack of realistic user motivation
- Bias toward initial usability
- Restricted spectrum of scenarios
- Reliance on simulated data
- Focus on individual usage versus the social and organizational context
- Neglect of mobility

The following sections cover each of these in turn.

54.2.1.1 Lack of Realistic User Motivation

Tasks in a conventional laboratory-based usability evaluation are typically based on some premise or assumption about motivation or goal, which the user is asked to accept. The evaluation itself is not designed to provide direct evidence about what the goals of real users are. By using an assumed motivation as the starting point, such evaluations effectively attempt to abstract usability from motivation. However, it stands to reason that user motivation interacts with factors that determine usability, *both objectively and subjectively*. In the laboratory, it is extremely difficult to simulate the level of emotional investment in the task that real users feel and that lead them to undertake the task with varying degrees of commitment in the first place. How people approach a task, how tolerant they are of difficulties along the way, and how they judge the ease of the task after the fact will be influenced by things like how urgent the task is to them, how much other pressure they are under from other tasks, what their motivation is to accomplish the task, and crucially, how much benefit they expect and experience at different stages. Similarly, motivation will certainly affect persistence. In many cases, this may make the difference between whether you abandon the task a moment before discovering the solution, or persist until successful. These motivational factors could either exacerbate perceived usability or mitigate them, depending on the exact circumstances.

Next, the specific motivations of different users may lead them to take different pathways through the interface, thereby encountering different problems. In real life, users do not necessarily have the single goal that a laboratory task asks them to adopt. They may have clusters of goals with complex relationships among them. It is quite possible that nuances of the specific motivations of real users might lead to taking a different route through a UI, so that they encounter a different subset of usability issues from that experienced by simulated users, whose route may be more externally determined.

It stands to reason that there is a complex relationship between the usability that you experience and the benefit you expect or actually obtain from using something. Obtaining meaningful benefit often depends on time, sometimes requiring extended usage. If you persist long enough to experience major benefits, you may see that the effort was worth it and not judge the experience negatively. For example, the value of reporting functions in business software partly depends on how well populated the system is with data, which can take time. In the "real world" (as opposed to the laboratory), motivation and expected benefit may also determine whether you even attempt to use functionality that you reasonably expect to be challenging or complex, but which could open up potentially rewarding capabilities of the system. None of this is meant to imply that we should not strive to reduce ease of use obstacles for experiencing benefit, simply that benefit is separate from and interacts with ease of use.

54.2.1.2 Bias toward Initial Usability

Most definitions of usability (e.g., ISO 9241-11 1998) include efficiency of use. However, laboratory usability usually, and perhaps inherently, deals with people who are unfamiliar with the system being tested. This means it tends to be most sensitive to problems of initial discovery and is not particularly sensitive to the issue of efficiency of use by experienced users with stable, established usage patterns. Even if initial ease of learning some aspect of the UI is low, a benefit of mastering it might be a great increase in efficiency, thereby ultimately producing an eventual net increase in usability that would not be detectable for some time.

In a sense, expectation or awareness of the existence of a tool's capability is the first phase of the discovery process. At the same time that laboratory usability evaluation may emphasize discoverability, it can paradoxically introduce a bias, because simply presenting the task scenario to the user can imply that there is in fact some way of accomplishing the task. Sometimes the more important question is whether in a natural and unguided usage situation it would even occur to a person that the capability exists, so that they would look for it spontaneously.

While usability is assumed to include learnability, laboratory usability testing with novice users does not directly study the learning curve that will occur with repeat usage. We tend to assume that we can generalize from initial use in the laboratory to use over time, that things that are the hardest to discover will be the hardest to find in the future, that the hardest things to figure out initially will be the most likely to be forgotten, and that the tasks with the most initial errors will tend to be most error-prone over time. Although these sound like reasonable assumptions, it is easy to think of situations in which they may not be true. It is quite possible that things that require the most effort to learn will be the most deeply imprinted and that a wide range factors affect usability nonlinearly over time. Usability obstacles of different types may emerge at different stages of the learning process, but will not be likely to be detected in a single laboratory session with users new to the current design.

54.2.1.3 Restricted Spectrum of Scenarios

In the laboratory, we provide a limited sample of the spectrum of experiences a real user will have with whatever we are testing. We take tasks out of the context of other tasks. This may provide the test user with a very distorted experience. In our experience, many usability disasters have been driven by the desire to make all things equally available to the user. In real usage, it is normal for the user to expect some functionality to be less obviously available than others, but assessment of this will be limited if the user does not have the opportunity to experience a realistic range of usage scenarios and thereby to form realistic expectations of what should be prioritized. Also, the user in the laboratory can only guess at what his or her real life priorities will be in the real usage context.

54.2.1.4 Simulated Data

How easy something is to use depends partially on one's familiarity with the universe of data that populates it, and how that data is structured in one's experience. Different information terminologies and architectures can evolve within the same domain, often for historical or even somewhat arbitrary reasons. The use of simulated data in usability testing can, therefore, introduce a serious confound. It can be very challenging to separate inherent usability problems with the design from problems that result from a sense of disorientation that comes from working in a universe of unfamiliar data. Usually, usability researchers try to make sure that the data with which a test UI is populated do not include distracting cues. This can be extremely difficult to ensure, but even using totally neutral data, if such a thing exists, may not be enough. In real life, familiar data might provide important landmarks that help the user navigate the system. A simulated environment may be highly discrepant from that of users in their real lives, and it is almost impossible to tease out this influence.

54.2.1.5 Individual Usability Abstracted from the Social and Organizational Context of Use

Laboratory usability evaluations typically focus on individual users. However, usability of many tools depends on organizational and social issues in the usage context. This is obviously true for collaboration tools, but applies to many other types of tools that have to be implemented within a larger system or that are used by individuals in an organization. Ensuring that the tool is usable in principle for an individual user as measured through test tasks may still not address whether it is usable in the organizational context of work flow, technical infrastructure, and so on. Usability problems may well exist at the organizational level. This adds another layer to the interaction between benefit and usability discussed earlier. The process of implementation and adoption of a system in an organization typically takes time. The opportunity to experience the eventual benefit of the system is in the future, and the road map toward it may be more or less visible to users of different types. Many systems do not provide their value until they have been implemented and widely adopted. Therefore, for such systems, we need to understand the balance between usability obstacles, experienced value, and anticipated value at various stages of the organizational implementation and adoption process.

The usability that users experience in real life can be the result of the composite efforts of the designers and of intermediaries within the organization. The ultimate end user experience of many business applications is partly the result of the efforts of people who customize the system for a specific installation and then populate it with data that embodies an information architecture the software designers may not control. The challenge of providing software that is adaptable to a wide range of situations means that design teams have to allow modifications and customization. They may try to build in constraints on what local implementers can do, but their control over the usability impact on end users may be very indirect. From the users' perspective, though, it matters little at what level the usability problems are introduced into the system.

54.2.1.6 Neglect of Mobility

Mobile usability is also a priority area for naturalistic evaluation, although the logistics of evaluating mobile devices in the field are certainly great. However, not only do environmental conditions, such as noise and lighting, play a major role in mobile usability, but the varying social contexts of use also have a major influence. For location-based services, the value of evaluation in context seems even more compelling (Goodman, Brewster, and Gray 2004). Despite the apparent value of field usability studies in the mobile domain, a literature review by Coursaris and Kim (2007) found the majority of mobile usability studies were conducted in the laboratory and called for more emphasis on field research.

Duh, Tan, and Chen (2006) compared results from a usability evaluation of mobile phones in the laboratory versus in the field. Using a variety of self-report and behavioral (e.g., task time) measures, they detected different, more numerous, and more severe usability problems in the realistic environment. Although one can raise questions about their methodology, such as whether carrying out a structured evaluation in a public setting introduced a global negative bias for participants, their findings highlight the likelihood that evaluation in the natural setting may produce a very different view of usability issues. Nielsen et al. (2006) did a study in which they similarly tried to isolate the effect of setting by using the same test protocol in the laboratory and in the field. Their study seems less subject to the concern that testing in a public setting was a serious confound, because of the specific setting used—a warehouse as opposed to a commuter train. They, too, found more usability problems in the field, as well as problems in categories that the laboratory was not sensitive to (e.g., cognitive load). Kaikkonen et al. (2005) did a similar study that led them to question the value of field evaluation over laboratory evaluations, on the grounds that they found the same list of problems in the laboratory as in the field. However, they used a larger sample than is common in iterative usability research (20 in each condition), increasing the likelihood that their list of problems uncovered would be fairly exhaustive. Also, overall problems occurred more frequently in the field, suggesting that the field may have indeed provided a more sensitive test. It is worth reiterating that none of these studies used a fully naturalistic approach, in that they used the identical, predetermined (rather than personalized) task scenarios in both conditions. A fair comparison of naturalistic and conventional laboratory evaluation would have taken into account the advantages of testing a mobile telephone under conditions where the participants actually care about their activities (e.g., communication) using the device.

54.2.1.7 Summary of Contrasts with Conventional Laboratory Evaluation

In contrast with conventional laboratory evaluations, studying usability in context allows for exploration of goals that matter to the user, of dependencies on contextual factors, and of the complex interactions among motivation, expected benefit, experienced usefulness, and usability. It eliminates confounds such as those attributable to simulated data. It provides a more comprehensive view of usability than is possible in a laboratory because naturally occurring task scenarios tend to be more diverse. It also allows for a more comprehensive view over time because it can take a broader look at the facilitators and obstacles in the product design that influence various stages in the evolution of the user's experience over time. To illustrate this, consider the following proposed model of the stages, or hurdles, in the transition from the existence of a user need to full adoption of a product.

1. Need: How does the user experience desires, goals, challenges, objectives that could motivate goal-directed behavior, or change in current goal-directed behavior, to which the tool is intended to be relevant? How well does the product address these in principle?
2. Expectation, seeking: To what extent is the user looking for a technical method for fulfilling the needs? How established are their current practices to fulfill the need, and how satisfied is the user with those practices, and how aware that there might be an alternative?
3. Exposure: How does or might the user gain the knowledge that a potentially relevant tool exists? What are the channels through which and the conditions under which this exposure takes place? How effectively does the process of exposure reach the most likely target users?
4. Discovery, recognition: How does or might the tool convey to the user its relevance to the user's need?
5. Exploration: How does or might the user approach the tool and how does the product engage the user to communicate what it might do for him/her *in principle*? To what extent does the product deliver on the potential, and does the user experience an increment in value to motivate continued experimentation?
6. Experimentation: What is the user's experience during initial attempts to apply the capabilities of the tool to specific instances of need to evaluate its relevance and potential benefit to him/her? To what extent does the user's initial experience encourage continuing broader experimentation with the tool?

7. Incorporation, adaptation: How does the user modify other related practices to allow for repeated, ongoing use of the tool? What is the burden of making these modifications? How does the user experience accumulated benefit? Is it sufficient to encourage sustaining this process?
8. Adoption: How does the user modify other related practices by incorporating the outputs of using the tool, building it into a network of practices that sustains utilization?

Any of these questions can be the target of evaluation research. Admittedly, this model does not apply equally to all interactive products. For example, the later stages are not as relevant to some tools that can be judged adequately based on the quality of the single-use experience. Although it is rare that a single naturalistic usability study would cover all these stages, the approach is flexible enough to allow choices about what steps in the model to emphasize and to allow varying the approach and focus across participants in a single study who may be at different points in the sequence of steps. For example, it allows for adapting the protocol to study the very different user experience issues that are relevant with people who are already users of a website and with people who have never used the site for that particular purpose.

Laboratory usability tends to focus on steps 4 and 5. On the early side of the process, naturalistic usability evaluation allows space for the researcher to gather more direct evidence of the degree to which a user spontaneously seeks a capability within the tool to meet some identified need, before prompting him to try it out. Working in the user's context allows time to investigate issues relevant to stages 1–3 as well. On the later side of the process, for example, stages 6–8, naturalistic usability can investigate facilitators and obstacles that arise beyond the first trial use of a feature and that influence the degree to which the user moves toward stable adoption of the tool for some uses.

54.2.2 Naturalistic Usability Compared to Other Types of Contextual User Research

When usability was a relatively new practice, evaluation was often called in when a product was thought to be far enough along in the development to give users a high-fidelity experience of it. Unfortunately, the result was that evaluations often turned up problems that could not be solved simply or elegantly at the level of screen design, but only at a deeper level. The experience of discovering these deeper problems has sensitized many usability professionals to the need to become involved in the design process earlier. Testing of prototypes in rapid paper prototyping (Snyder 2003) and the RITE method (Medlock et al. 2005) are examples of responses to the need to incorporate evaluation early and iteratively during the design process.

But studying early design ideas is not enough. As discussed in Section 54.2.1.7, to ensure that products are not only designed properly, but also play appropriate roles and fit into people's lives, requires work in the usage context to understand user characteristics, goals, and needs that users bring with them as well as environmental factors like work practices, organizational factors, technical infrastructure, to name a few.

The very issues that tend to be neglected in laboratory usability testing have over the years become the focus of a group of approaches that focus on understanding user needs, motivations, goals, contextual constraints, and influences. No standard nomenclature exists to describe this group of approaches. Some people have simply expanded the meaning of "usability" so that it now incorporates issues of usefulness, experienced value, desirability, hedonics, contextual fit, and so on, rather than simply cognitive usability, or have referred to this broadened meaning as "Big U usability" (Barnum 2011). However, this potentially blurs the still meaningful distinction between the traditional narrower definition, focused on design attributes that facilitate cognitive and task performance aspects of the user experience, and this broader concept. We prefer the terms "User Centered Design Research" or "User Experience Research" to encompass the spectrum of research that attempts to facilitate designs that achieve the wider range of virtues we strive for beyond "small-u" (or traditionally defined) usability. These terms cover the very broad spectrum of approaches for introducing data from and about users into the design process. Among the most important of these are research techniques that study the behavior of users or intended users in their natural contexts to get at more fundamental issues that should influence design at deeper levels. Here, we address two main forms of this research, ethnography and contextual inquiry. Both these are discussed far more comprehensively in Chapters 43 and 45. The purpose here is only to contrast them with naturalistic usability evaluation. These contrasts are admittedly more subtle than the contrast with laboratory usability.

54.2.2.1 Ethnography

Ethnography was introduced into the world of product planning and design from the field of anthropology, as an approach to introducing deep knowledge of users into the product planning and design process. Just as with usability, there are some issues of nomenclature in regard to ethnography. As ethnography has become increasingly accepted in industry, more people have adopted (or co-opted) the term to describe what they do, to an extent that its meaning in practice can become diffuse. Nowadays, "ethnography" is too often used loosely as if it covers any research about users in their natural environment, even simply interviewing them on-site. The following paragraphs propose what is hopefully a more useful way of thinking about the distinctive character of ethnography, emphasizing things that differentiate it from naturalistic usability.

Ethnography as practiced in industry, or "design ethnography" (Salvador, Bell, and Anderson 1999), tries to understand the context of basic human and social functions within which a successful product must fit to help ensure that the product plays a meaningful and useful role. It tries to focus on a more basic level of human activity that is not tied to the particular tools currently used to fulfill those functions. For example, while a usability evaluation of a website might examine whether people can find the information they want regarding a product, an ethnographer might try to understand the concept of "confidence" in a purchase decision and the social dynamics surrounding it. Information may play a role in helping a person develop confidence in a service or product offering during online shopping, but any one piece of information or one website feature contributes only one small piece to the larger picture of "confidence." Understanding what is at stake for the individual in making a major purchase decision, what causes anxiety, what the balance is between rational decision making versus justifying an emotional decision, what the social dynamics are around establishing confidence of others in the judgment of the shopper and gaining their support for the purchase, and so on may all inform overall product strategy and help ensure that implicit levels of the user experience are aligned with these dynamics across many aspects of the design.

This focus on fundamental understanding can create a challenge for ethnography in industry because its implications for specific product-related decisions can be perceived as too indirect or abstract. Too often, the implications are expressed at a level that leaves room for a great deal of interpretation and inference on the way to a specific implementation. Product teams looking to ethnography for design guidance sometimes complain that the research gives them an interesting and vivid picture of the user, but leaves them guessing about the concrete product decisions they have to make. This issue has received a great deal of attention from the design ethnography community (e.g., Diggins and Tolmi 2003; Beers and Whitney 2006; Jones 2006). Design ethnography, done well, can have very powerful implications for products, but these are often at the strategic level rather than the more concrete design level of small-u usability.

54.2.2.2 Contextual Inquiry

Contextual inquiry (CI) is an approach to exploring people's interaction with technology by combining interview with observation while they are in the process of carrying out their work. It has been an extremely influential technique in user research. Beyer and Holtzblatt (1998) explain that it is based on the belief that only by grounding interviews in observation of actual work can you compensate for the weaknesses of self-report, such as biased recall, tendency to give summary information, and difficulty recognizing how one's own behavior depends on the context.

CI is not so much an evaluation of the tools observed, but an attempt to use observation of user behavior with tools, combined with a particular style of probing into the users' explications, to understand how users approach their work, to infer deeper levels of user goals and of how work is organized, and to suggest new design ideas. These ideas do not necessarily map in a direct manner to a list of solutions to identified problems (McDonald, Monahan, and Cockton 2006). Although CI certainly attempts to capture evidence of "breakdowns" or limitations in the usefulness of current systems by noting things like workarounds that users have developed, there remains a difference of emphasis between the typical contextual field study and naturalistic usability evaluation in how intensively they focus on strategies to elicit data that identify usability issues directly applicable to the new design.

One limitation of CI as an evaluation method is that it is normally carried out while people interact with their current, customary tools to achieve their goals. Experienced users have often adapted to some usability problems that might block early users. In a sense, this is the complement of a limitation cited above for laboratory evaluation—the focus on novice users and discoverability problems. Experienced users may or may not recognize ways in which their process could be more efficient or less error prone. Their work practices may have been designed around some of the main limitations of their tools so that they do not experience them as problems, even if these practices are in principle sub-optimal. Similarly, doing research with experienced users limits the researcher's ability to learn about usability obstacles that arise at different points in the discovery and learning process. If one plans the sample to address this, such as by including participants who have different levels of experience, or probes on less habitual tasks, the distinction between naturalistic usability and CI can begin to break down.

As stated in Section 54.2.2, the contrast between naturalistic evaluation and these two field research approaches is more subtle than that with laboratory evaluation. Focusing on the evaluation of technology in use makes it easier to derive concrete design implications than is usually the case with ethnographic approaches. This is also true of CI, but naturalistic usability evaluation also complements CI, because it does not limit the research to experienced current users of your tools, although it may include them. Also, while CI does allow for and indeed encourages probing of variations in people's task approaches under different circumstances, naturalistic usability goes beyond this as an evaluation method by encouraging people (judiciously and in a highly personalized manner) to undertake tasks and use features in ways they may not have done on their own.

54.3 METHODOLOGICAL CHALLENGES OF NATURALISTIC USABILITY

Doing naturalistic usability evaluations would be easy if all they involved was finding existing users, watching them do their tasks on their own, and seeing where they get into difficulty. Unfortunately, this is not the case. Intervention by the researcher is almost always necessary, at every stage from actively selecting a sample to exposing users to tools, to engaging them in tasks as naturally as possible, to probing their behavior and thinking during their task performance, to prompting their behavior. These actions obviously compromise "naturalness" of the research. This section discusses methodological considerations that the researcher must take into account to make sound decisions about these interventions and to manage their impacts.

54.3.1 Spectrum of Naturalistic Usability

We can distinguish highly naturalistic from semi-naturalistic approaches, with gradations between them, based on the amount of researcher intervention into the natural usage experience that they entail. A maximal degree of naturalism would involve almost no intervention and minimal interaction with the user. A minimal degree may involve a fairly structured usability evaluation in a simulated environment where the contents of the tasks are nevertheless highly personalized to the user. We can also make finer-grained comparisons among approaches based on choices about how to handle specific aspects or dimensions of naturalism. Semi-naturalistic evaluations may compromise on some of these dimensions while trying to preserve others.

In fully naturalistic approaches, the evaluation is done with people who have full freedom to use the technology as they see fit. They are intrinsically motivated to use it for a realistic spectrum of activities and to attempt to get value from it. The researcher has the opportunity to study their experience over sufficient time to understand the realistic spectrum of user-attempted tasks, and does not need to intervene in the natural process of usage, whether to prompt user behavior or even to ask probing questions to understand the user intentions. Unfortunately, situations where pure observation of completely unconstrained user behavior will provide adequate usability data are extremely rare, because of the type of information needed, practical constraints, and the fact that the researcher does indeed have an agenda.

Consider the following case situation where a fully naturalistic approach would make sense in order to appreciate how rare it is. It involved a single-purpose walk-up-and-use-system of a type that could be usefully evaluated through observation alone, without the need to interact with the user at all, namely, a self-service coffee machine. Its industrial design was beautiful. It looked like industrial sculpture, a robotic barista that took infinite care to brew each cup just for you. This beauty conveyed the promise of a very high quality cup of coffee. From the sounds it made, you could tell that it ground the beans, deposited them in some kind of receptacle, and injected hot water slowly over the ground coffee. About 10 seconds after the hiss of the water began, coffee would begin slowly dripping out of the nozzle. Probably because the process was so slow, it had been designed with a helpful LCD progress indicator (beautifully framed with polished brass trim) that would slowly fill from left to right. Three sizes of coffee were available. For a small cup, the progress bar would slowly fill once. For a medium cup, it would slowly fill once and stop when the cup was half full. After a slight pause, and some additional mechanical noises, the progress bar would start from the beginning a second time, and more coffee would begin dripping into your cup. For a Super Grande, the process was repeated three times. The problem was that, having waited so long for the coffee to brew, people had no idea what was going on when, although their cups were only half full, the progress bar seemed to give them the message that the process was finished.

Now a piece of context: the machine was in a cafeteria in a conference center. Naturally, most coffee was ordered at busy times, with a long line of people waiting, most of whom were encountering the machine for the first time. Any usability researcher could stand nearby and watch as one hapless user after another took his half empty cup from under the nozzle, stared at it in puzzlement, noticed with confusion that the machine had started up again all by itself, and thrust his cup back under the nozzle a moment too late when he realized that the liquid now dripping into the drain was the rest of his own order. The waiting customers (most of whom *were* usability researchers) could see that something was going on, but could not see what the problem was, and so repeated this experience one after another.

This case had a number of characteristics that would enable adequate understanding with a very simple methodology and make the cost of research per user studied very low.

- Users were motivated by their own goal without any need for artifice or special recruiting efforts.
- There was no need for them to explain their intent to the researcher, and their interpretation of the system feedback was obvious.
- The initial usage experience was by far the most crucial one (since a large percentage of users were going to be using the machine only once).
- The time needed to observe each user's complete experience was short.
- There was a naturally occurring concentration of users in time and space who provided a good sample of the spectrum of usage scenarios (e.g., small, medium, and large cups; regular coffee vs. cappuccino, etc.).

These are the opposite of the circumstances that apply in most naturalistic usability studies, however. It is far more common that the researcher needs to compromise some aspects of naturalism by applying active management strategies to ensure participant motivation and to balance user freedom and spontaneity with the goals of the research by intervening in the user's natural process of interacting with the tool. Any intervention on the part of the researcher introduces some degree of artificiality, but the goal of naturalistic usability evaluation is to minimize this by carefully "titrating" the researcher's influence. The rest of this section discusses methodological considerations for managing user motivation and user freedom and concludes with discussion of the special issues related to these topics in the case of longitudinal studies.

54.3.1.1 Ensuring Motivation

The primary dimension of naturalistic evaluations is user motivation. All naturalistic usability evaluations embody some strategy for achieving as realistic a degree of motivation or engagement on the part of the user as possible. Engagement is influenced by how committed the test user is to the goals that the tool being evaluated is supposed to support. This does not necessarily mean that naturalistic evaluation enforces this commitment either to the goal or to using the tool, because one of the research questions may be whether the tool itself or any feature of it (or any information about the tool the user may be exposed to) evokes motivation.

The design of the evaluation study can influence this by controlling such things as whether the user has actually made a personal investment in the tool being evaluated, expects to live with it for an extended period, sees its use as mandatory versus optional, and/or has access to appealing alternatives.

However, the agenda for most studies goes beyond assessing how effectively the tool as a whole attracts people to use it. More often, we want to assess the experience the tool provides to people who do engage with it, including how attracted they are to using components of it. Realistic motivation can be incorporated into a research project in a variety of ways. The specific approach depends on the goals of the research and the constraints of the available research opportunity. A usability evaluation of a newly implemented work system, use of which is not optional and that employees expect to have to live with for an extended period of time, might be highly naturalistic depending on how the researcher structures the interaction with participants. Another example would be an evaluation of a product with a user who has actually purchased it. An "out-of-box" evaluation (OOBE) with an actual purchaser of a new product, that is, setting it up for the first time in the intended usage environment, is a simple example of such a study. An intermediate level of naturalistic motivation can be represented by an evaluation during an agreed-upon trial period that is structured so that the person has a strong incentive to get meaningful benefit from the tool. For example, in a study on the experience of switching to a new e-mail tool, we recruited people who agreed to adopt the new tool for an extended period. Another step down in naturalness might be a situation in which participants use a tool that they are not necessarily expecting to adopt for an extended period, but that they use for test tasks that nevertheless do reflect their own goals. An example would be a test of a website where the user is attempting to get the information that they are actually interested in at the present moment. Finally, an evaluation of a system that the user is interacting with only for limited test purposes and for a limited number of task scenarios can still be considered somewhat (or minimally) naturalistic if the tasks are highly personalized to the participant, so that the participant cares about them. This can be achieved if the user has the possibility of deriving some meaningful benefit even from a brief interaction.

Even though the user's engagement should be determined by realistic motivation, project timelines typically require us to obtain data in concentrated form within a limited time frame. Therefore, a naturalistic study may require either a recruiting strategy for finding a sample of people who are naturally motivated to do a task at a particular time convenient for the research or finding people who have some meaningful pre-existing motivation and then inducing them to do the task at a particular time.

Obviously, the former is more "natural." In some studies, identifying such a group may be fairly easy: working through a hospital or clinic to recruit a group of patients who have just received a particular medical technology; partnering with a retailer to recruit a group of people who are new purchasers of a particular tool; studying the user experience in companies where a new system has just been implemented and large numbers of people are going through the migration process at the same time. Sometimes the study can be scheduled in a season when the behavior of interest is common in the population, making it easy to find users who are ready to proceed, for example, evaluating a gardening website in the late summer and early fall when local perennial gardeners are likely to be planning their fall planting, or evaluating a tax tool in the United States between January and April.

On the other hand, often the best one can do is to identify people for whom there is evidence of likely motivation but who are not yet engaging in the behavior of interest. E-commerce studies can often recruit people who are currently "in the market" and, therefore, can realistically engage in online shopping or even purchasing during a research session. An example might be recruiting people for a travel website study who are expecting to take a vacation in the next 3 months, who have not yet finalized their plans, and who expect to use the Web to make a reservation and arranging for them to do so during the study. One strategy to induce this behavior is to give people the incentive for participation in the form of reimbursement for or contribution to the purchase they make during the study.

Finding people who are at the appropriate point in their own process to fit the focus of the evaluation can be challenging. The recruiting itself can be done using a traditional screener or by selecting people based on some behavioral indicator of motivation. For example, people browsing in a certain department of a store may be recruited for an online shopping study for that type of product; families visiting college campuses may be recruited for a study of an online tool for college financing.

In any form of usability research, we should always consider biases that may be introduced by the recruiting and selection process, and this is certainly true here. In particular, it is important to acknowledge that people who are at a point where they are ready to take the action of interest to the researcher on their own may be different on average from people who are at an earlier stage and need to be induced to act. A sample of people who are shopping now but not quite ready to purchase on their own will include a certain percentage of people who would not otherwise follow through to an actual purchase. The researcher must pay attention to how stringent the selection criteria are regarding evidence of pre-existing motivation versus how strong the inducement to engage in the behavior needs to be. In some cases, one can get a quantitative sense of this by finding statistics about how strong a predictor the recruiting criteria or targeting strategy are of the behavior of interest. However, even in cases where only a qualitative assessment of this is possible, it is important to make it explicit.

54.3.1.2 Freedom

Naturalistic usability evaluation involves a range of freedom to interact with the technology in an unconstrained way. It can also vary in how much the research depends only on the tasks the user undertakes spontaneously. Most naturalistic studies require some degree of prompting to get participants to engage in particular tasks. The degree to which this is needed depends on the degree of spontaneous motivation in the sample the researcher is able to assemble and on the amount of time the researcher can allow during the evaluation for users to proceed on their own initiative.

Obviously, manipulating the timing of a task by inducing people to do something during the study that they may be close to doing on their own is one form of intervention in the user's natural behavior. In addition, the researcher often has to induce the person to use a tool that is different from what he or she would have chosen spontaneously. For example, people shopping for a cell phone might be given a specific phone to live with for evaluation purposes, even though they might not have chosen this one on their own. At a more micro level, even a naturalistic study typically requires some form of prompting for people to follow a different usage path than they would have done on their own, such as to use a feature or attempt a specific task. To the extent that the users are still guided by their own goal and that the task outcome matters to them, the study can still be considered at least partially naturalistic.

The degree of user freedom in a naturalistic usability study requires a special approach to scripting of the research protocol. A conventional laboratory usability study can be tightly scripted. Of course, room must still be given for users to construe some aspects of the task in their own way and to choose their own approach to attempting it. This is where the data comes from, after all. However, scenarios can be written and pilot-tested to ensure that the portions users "should" understand in a similar way are communicated clearly, so that the givens of each scenario (the starting point in the UI and the task assumptions) are standardized. The sequence of scenarios can be predetermined and assigned to users in a predetermined or rule-based way.

The situation is very different in naturalistic usability research, because the balance between freedom and scripting is shifted heavily toward freedom and open-ended exploration. This can shape even how the session begins. Laboratory usability sessions may start with a brief interview that includes some open-ended questions. These provide individual information beyond what is available in the screener data. Both the screener and interview data may help the researcher to interpret the user's response to the scenarios or to omit scenarios from among the pool that seem to be less applicable to the user. In a naturalistic usability evaluation, this individual background investigation is much more extensive and may take on the character of an extended interview or even an ethnographic investigation with each participant. The purpose is to understand as much as possible about the context and baseline practices of each user, not only to facilitate interpretation of the participant's subsequent behavior when interacting with the technology you are evaluating, but also to support deep customization and personalization of the evaluation process.

When work begins with the tools being evaluated, users are given much more freedom to define their own goals and tasks. The process of interacting with the user may look very similar to CI, with the difference that the user is not assumed to be an expert on the tool and that the researcher has a primary agenda of potential scenarios that focus on evaluation. If the researcher suggests tasks, they are typically delivered in an open-ended or general manner, with the user free to add concrete details that make the task personally relevant. Alternatively, the researcher may improvise task instructions in more detail based on the knowledge of the user's specific goals and self-defined tasks.

Decisions about prompting the user represent a compromise between some predetermined priorities for the research and the specific opportunities each user presents to probe different aspects of the interaction design. Because the researcher has more information about the user as the session (or, in the case of a longitudinal study, the sessions) proceeds, it becomes possible to make more and more nuanced decisions about how to balance the research priorities with the opportunistic aspects of the process. Therefore, the degree of direction and structure that the researcher provides typically increases during the evaluation.

The idea of the researcher prompting the user raises concerns about researcher intervention contaminating the findings. There is no perfect solution to this problem. It cannot be eliminated, only "managed." When prompting the user to attempt a task, the bias should normally be to start with as little direction as possible to allow the opportunity to observe the user's natural exploratory behavior, as a clue to motivation and interest. Similarly, once the user is engaged with a task, the process of probing or giving cues should proceed in small increments, as in any formative testing. This gives the researcher a chance to see what the minimum level of cuing is for the participant to move forward and to experiment with different cues to see which make the difference. Of course, the sequence of prompting must be carefully documented and used to help interpret subsequent findings. This information can provide clues about what cues the design must give to facilitate any step in the process of engagement and adoption, from leading the participant to consider exploring a tool or feature, to applying it to their needs.

The open and improvisatory character of naturalistic evaluation requires very different preparation from what goes into highly scripted laboratory evaluations. Typically, it is important to prepare a list of general categories of usage scenarios of interest, described in a fairly generic manner, to enable the researcher to recognize relevant scenarios that arise with each participant. These need to be matched with parts of the tool expected to play a role in supporting such scenarios. Identifying potential areas of design concern will help with prioritizing these. This is not inherently different from the process of choosing what you want to test in a laboratory evaluation, except that the list of priorities will often be much longer, because you need to be prepared for the wide range of opportunities the participants will provide based on their differing motivations and task approaches.

54.3.1.3 Special Methodological Issues for Longitudinal Studies

To this point, we have been talking about naturalistic studies as though they provide a snapshot of usability at a single point in the user's history with the tool, either with people having their initial experience or for people with more experience. But what if we want to understand the experience over time, for people with different amounts of experience? The choice is between cross-sectional versus longitudinal research. Cross-sectional research requires studying separate sub-samples that reflect different amounts of usage experience. It provides a series of snapshots. The problem with this approach is that the samples for the separate snapshots cannot be assumed to be equivalent, because of differences among the people who began their experience at different points in time (e.g., the "oldest" group might include more "early adopters" or may have experienced earlier versions of the tool) or because of nonrandom dropouts along the way. This may be less true with some business systems where usage is not optional, but to the extent that the samples are nonequivalent, the researcher will have difficulty teasing out this confound, which is known as "survivorship bias."

In contrast to cross-sectional research, longitudinal research follows a given sample over time so that the evolution of each individual can be tracked. This approach has many challenges of its own. The demands of participation that will extend over a longer period may make it harder to find willing participants, and thus, introduce a selection bias. In any sample followed over time, there will be attrition, with people dropping out for various reasons. Some of these reasons may be due to highly individual changing circumstances that make participation in the study difficult and are, therefore, in a sense "accidental" or random. But some of the reasons may be correlated with user experience issues. In many real life situations, people are free to abandon a tool to return to their previous one or a new one based on their experiences. As mentioned earlier, this may be less true in some business situations, but even in business settings people often have some freedom about how exclusively they rely on a tool and how many of its capabilities they explore and adopt.

In contrast to cross-sectional studies, longitudinal studies allow the researcher to obtain more direct evidence of what leads to the dropouts, which can be used to interpret the difference between the original sample and the end sample of survivors. However, it adds the problem of ensuring that you end up with enough people going far enough with the tool you are evaluating to cover deeply enough the design issues you want to explore. It is often hard to estimate in advance what the natural dropout rate will be. In longitudinal epidemiological studies, researchers trying to identify risk factors prospectively typically start with very large samples because they know that only a small percentage will develop the disease (and because of the number of variables they want to screen), but we do not have that luxury in user experience research, where we typically have small samples. In addition, in longitudinal research, there is a particularly large investment in each participant, so dropouts are costly.

This creates a strong incentive to try to retain participants who might otherwise have abandoned using the tool or to help them over a usability hurdle that would have blocked further development of their usage behavior. Of course, this means that findings from these people beyond that point may be more speculative. Nevertheless, they can still provide insight into usability beyond what would otherwise have been their stopping point. Clearly, the researcher must treat the need for this type of help as an item of data and carefully document it, as described earlier.

There is a more subtle form of attrition to consider as well. If an individual's exploration of the tool and experimentation in applying its functionality to their needs level off quickly, the amount of information obtained from them, and therefore their value in the study, may be out of proportion to the amount of time invested. On the other hand, people who persist in exploration spontaneously may not be representative of the wider range of users. They may be different either because the natural variability in their work tasks makes a wider range of tool features relevant to them and creates more opportunity to use them, or simply because of their own curiosity, comfort with exploring technology, and so on. Therefore, it can be necessary to push users beyond their existing usage behavior by negotiating usage experiments with them, incorporating content and addressing goals that are potentially meaningful to them. Again, it is important to document these decision points with each user.

54.4 CASE STUDIES

This section provides three case examples of naturalistic usability studies in the field. Each case includes a discussion of some of the findings, to show the particular benefits of the method, and discussion of the methodological decisions that had to be addressed for each study.

54.4.1 E-Mail Switching and Adoption

We did research for a client who offered an e-mail service as part of a collection of other web content and features available in both free and subscription packages. Users could adopt the e-mail as a stand-alone service or as part of a larger package of other services through a web portal. The client had developed specific tools to facilitate migration from a person's existing e-mail into the new service, such as tools to import contacts and existing e-mail folders, and to notify contacts of the person's new e-mail address. The client wanted to understand the usability of these tools and their contribution to the appeal of switching e-mail providers. We wanted to understand usability of these features in the context of how people actually set up their e-mail folders and managed their contacts. Also, the experienced value of the e-mail service had to be understood in relation to the experienced value of all the other services a person might encounter when adopting the new e-mail.

We discovered that people used the opportunity of switching as a time to "clean out" their old e-mails and existing contacts. This meant they needed a more convenient way of selecting items at a range of levels of granularity, rather than moving things en masse. This is something that could only be discovered by testing these features by having people manage their own data, since these are the data that they actually cared about.

We knew that a fair test of the e-mail service would require an extended time. For instance, one can only evaluate how well spam filter works when one has accumulated a reasonable experience base with it. Similarly, we needed to allow sufficient time for the process of discovering, exploring, and experimenting with various features both in the e-mail tools and in the associated web portal. This meant that we needed to recruit people willing to commit to the e-mail tool for an extended period. We gave them free subscriptions to the e-mail portal for 1 year and asked for them to use this new e-mail account and to commit to the research for a period of several months, with several visits scheduled during this period. Not only did this time commitment allow for the emergence and stabilization of usage patterns, but it also promoted their commitment to getting the best value out of the tool, since they were going to live with it for a while.

The demands of the study required that we recruit people willing to switch e-mail providers and that we pay them an appropriate incentive. This of course raises a question about the representativeness of the sample. We never conceptualize qualitative studies like these as statistical studies in which we are trying to find the average of certain variables and generalize it to a larger population. However, it is inevitable that the question will arise about whether our qualitative findings are applicable to real users or real likely users. For example, maybe the only people who would agree to switch their e-mail for research purposes were people for whom personal e-mail was not very important. To address these questions, we not only screened people to get a sense of their e-mail and web usage before recruiting them into the study, but also began the field research by doing an ethnographic study with participants to understand their baseline usage pattern and to make sure that the sample included a good spectrum of users. This of course also provided us with historical context for each participant to understand how e-mail fit into his or her life.

54.4.2 Real Estate Website

We did a series of studies on a website that allowed users to search or browse for homes listed for sale in a variety of ways, to identify real estate agents, request showings, get information about real estate-related services such as mortgages, and so on. The site allowed users to compose searches by selecting from a rich set of search criteria. They could save these searches and specific real estate properties located by the searches. They could set up e-mail notifications so that they would find out when new properties came on the market that met their criteria.

Contrasting two of the studies from this series will highlight the special challenges and benefits of naturalistic usability evaluation. The first study was a classic laboratory usability study. For this, we recruited users who were “like” real estate shoppers who might use a website like this. To be sure that any problems were not due to general lack of familiarity with the Web, they had to report a certain amount of prior online experience. To be sure that the kinds of terms and issues relevant to shopping for a new home would be meaningful to them, we required that they had purchased a new home within a certain time frame or anticipated doing so within a certain future time frame. Our client had commissioned the study because they had some very specific concerns about the understandability and navigational structure of the site. The task scenarios were, therefore, constructed to probe these issues, which meant that we prescribed task goals and specified certain other task parameters to the participants. All these were appropriate for exploring cognitive issues that might be inherent in the design in a way that was divorced or abstracted from the real context of use by people in the real estate market right now.

We learned many interesting things, for example:

- People were confused by search terms that overlapped in meaning and seemed to produce inconsistent results. For example, people were confused about why they would get fewer results when they searched for houses with a garage than when they searched for houses with a two-car garage. The answer had to do with problems in managing the process of entering property descriptions by real estate agents.
- People had problems dealing with the logical heterogeneity of the list of geographical areas they could use to filter searches. For example, when trying to filter a geography using a list of locations that included the names of cities, neighborhoods within cities, and towns, they were unclear how they would get to see properties in unincorporated areas of the counties that contained these towns and cities. If they selected the county, depending on the pattern of results, they were often confused about whether their search included results from the entire county *including* its cities and towns or if the results were only from the parts of the county *outside of* the listed cities and towns.
- When trying to decipher the underlying logic of the search options by testing them, they were often confused about whether the problem was with the logic of the classification, the accuracy of the search results, or something they may have done.
- When saving a search, they were often confused about whether they were saving the search parameters or the particular results the search had retrieved on that occasion.
- If they wanted to return to the search later, people looked in the “Search” tab rather than the “My Portfolio” tab, an ambiguous name that had been chosen because it subsumed both “saved properties” and “saved searches,” both of which would have been more concrete and understandable had they been easily visible.
- When given a task that asked them to set up notifications, they had difficulty finding the functionality; did not understand the relationships and

dependencies among saving a search, setting up a notification, and creating an account; and had difficulty navigating the interconnected processes for these functions.

To find out how to overcome these and other issues, we did a contrasting follow up study in a far more naturalistic manner. This time, the need to assure realistic, natural levels of engagement by participants was addressed by a different type of recruiting process. Through a network of real estate agents working for the large agency that owned the website, we identified a pool of people who were currently shopping for a home, from which we selected a sample covering a range of levels of engagement with the website. We accepted the tradeoff that people who already had a relationship with a real estate agent were probably more deeply engaged in the shopping process and would not be representative of people who might use the website while in an earlier or less serious stage of considering a move or a purchase (and took this into account in our interpretation).

We visited participants in their own homes or, for some people who were relocating to the area, in their temporary apartments, and explored the role that use of the website was playing in their larger process of house-hunting. For example, we were able to examine the role of the website in relation to their interactions with their real estate agent. By studying their use on their own computers, we were able to examine things like how they supplemented the website with handwritten notes, and what notifications they had received in e-mail and what they had done with them. We could examine to what extent and how they had used the many features of the website, and therefore, we could better understand the motivations that led them to these features in the first place, as well as the usability issues they encountered in the context of the usage scenarios they had attempted on their own. Many of these spontaneous scenarios were very different from what we had investigated in the laboratory, and they revealed a set of issues with a very different character. Finally, we could identify features of the website that held potential value for them, but that they had not discovered or had explored and not adopted.

We managed the balance of participant freedom and researcher direction through the process of data collection by combining elements of ethnography, contextual inquiry, and highly personalized usability evaluation. We began by exploring the participants' overall experience of house hunting and considering a purchase and a move, as well as the role of the website in the context of that broader experience. We studied their relevant artifacts—the traces left on paper or on their computers of their previous house hunting activities. When we turned to the website, we asked them to pretend that the visit had just happened to coincide with a time that they would have ordinarily done some online real estate shopping or research in whatever way they would typically do it. All the sessions involved some degree of prompting to attempt tasks on the website beyond what they had done on their own initiative. The selection of tasks included a mix of highly personalized ones to probe just beyond what each individual had already attempted, along with ones that pushed them to features we had our own questions about. With people who had less engagement with the website, we naturally used more probing to understand why this was the case and more active prompting to attempt tasks on the website that were relevant to what they were trying to do through other means. For all these "prompted" tasks, the content of the scenarios was improvised based on what we had learned about their own activities and interests.

Regarding the dimension of time, even though our visits were single sessions of 2 hours, we were able to get cross-sectional information and at least some retrospective historical information from them to get at least an indirect longitudinal view. Participants were naturally at different stages of the real estate shopping process (although all of them had at least moved to the point that they had a real estate agent working with them). In addition, their artifacts (e.g., e-mails, bookmarks, older saved searches, etc.) that we examined had often left traces from earlier phases of their shopping experience.

Here are some examples of the types of findings from this study that show the benefits of this approach:

- We identified a high priority group of users who were most likely to value the website but also most likely to have a particular difficulty. A surprising number of our participants were people transferring to the area. They had begun using the website before moving to temporary quarters while they continued their search. Clearly, this was a group for whom being able to begin their search remotely was a particularly compelling reason for using the website. However, for these people, managing the geographic scope of their searches was a particular problem, because of lack of the comprehensive local knowledge that the site assumed.
- Most people had not anticipated that a notification service was available and did not discover it. Once prompted to explore it, they saw its potential value, but then encountered the same usability problems we had seen in the laboratory.
- For those people who had discovered and begun using the saved search and notifications features, there were common problems with having constructed and saved logically overlapping searches, and so they received annoying multiple notifications on the same property, because there was no synchronization among searches.
- In trying to decide whether or not to save a search, people had difficulty determining whether a search that turned up few or no results was a "bad" search that should not be saved. They did not understand that, paradoxically, this was likely to happen when they had added many criteria, thus defining a very rare type of property, which was exactly the situation where saved searches and notifications would potentially be most useful.

- Not surprisingly, many people who were shopping for real estate were also selling their existing homes. Thus they turned to the website for different things, depending on whether they were wearing the hat of seller or buyer at any moment. We identified a number of ways in which usefulness and usability of the website could be improved to help them manage these two roles. As sellers, they were interested in different types of information on sold properties and needed better ways to keep their research activities as sellers and as buyers distinct.
- We identified different problems with the information architecture of search criteria (and the interaction with the underlying structure of the database and inconsistencies in data entry by listing agents) than we had seen in the laboratory, because in the field we were able to focus on issues with the types of criteria that were most important to them. We also were able to map some of these priorities and associated difficulties onto different user segments (stage of life, urban vs. suburban focus, etc.) in a way that would not have been possible with a sample of people who were not currently engaged in shopping.

54.4.3 Tablet PC Field Trial

Before the introduction of Microsoft's Tablet PC in 2002, we did a series of field trials in which groups of participants in several companies replaced their desktop PCs with functional prototypes of the new computer, which they used as their sole workplace computer for several weeks, while we studied the evolution of their experience longitudinally (Dray et al. 2002). The fact that participants needed to perform their real jobs ensured a realistic level of engagement with the device. The fact that they expected to rely on the devices for an extended period gave them a realistic motivation to learn how to use and try to derive value from it.

These studies were complex in many ways, including their logistics, their research agendas, and their data collection protocols. They involved longitudinal tracking of each user's process of discovery, experimentation, and personal experience and assessment of benefit, as they explored the capabilities of the Tablet and the experience of working with applications on this new platform. Investigation of usability was almost inseparable from investigating issues of user motivations, goals, and experienced usefulness or lack of it. The research protocol included a mix of the following:

- Ethnographic components to understand the work context and users' preexisting work practices
- Free observation of use of the prototypes under a wide range of work place situations, such as use at one's desk for a range of tasks and use in meetings
- Probing as users undertook tasks on their own
- A mix of opportunistic and highly personalized usability tasks closely tied to what users were spontaneously attempting on their own and somewhat standardized usability tasks following predetermined scripts
- Retrospective review of user experiences during the periods between visits, based on self report, logs of all service and support contacts, and walkthroughs of work artifacts

The research process involved a dynamic balance of user freedom and researcher direction. We had an extensive program of usability tasks that we administered in a planned sequence, tailored to fit the stages of the usage experience. These tasks varied greatly in amount of predetermined structure. Some of them were fairly standardized across users and were designed to evaluate things that were essential parts of the user experience, so that providing the task was not likely to bias the natural course of the evolving user experience. For example, an essential early activity for all users was learning how to use the stylus both to write on the screen and to activate on-screen controls (icons, menu items, links, etc.). We used standard tasks to test these things across users and repeated them at standard intervals to get a sense of the learning curve.

Other tasks were planned only at a generic level. For example, we knew we wanted to investigate the experience of converting handwriting to text and correcting conversion errors. We knew that handwritten input within dialogues, as opposed to free writing on documents, was a special case of this. The fact that people spontaneously and frequently attempted to do this in the "Save As" dialogue provided an opportunistic instance of this generic category of task. The fact that they did this at their desk and at meetings enabled us to evaluate the experience under different circumstances.

Finally, some tasks were highly opportunistic. Different users were naturally drawn to different Tablet features and to use different applications on the Tablet in different ways, according to their jobs and work practices. Therefore, different people provided different opportunities for evaluation. For example, one participant who was involved in contracting provided many opportunities to explore the use of handwriting to annotate digital documents. Mobility was more important to some participants than others. Specifically, since this study was done at a time when most people used desktop PCs wired to the network, we gained insights into problems they had in trusting that the document they opened on the Tablet during a meeting was the latest version that had been e-mailed to meeting participants. Highly mobile participants provided opportunities to evaluate power management issues. At a more granular level, people used different feature sets within applications, giving us a wide range of evaluation opportunities.

The fact that this was a longitudinal study in which there was very large investment in each participant (both in terms of research time and cost for things like the hand-built prototypes they used) intensified the issues around balancing user freedom with researcher intervention. We wanted to learn as much as possible about the natural and undirected process by which people would explore the capabilities and value of the Tablet. At the same time, we wanted to use the opportunity presented by this research to evaluate the Tablet as thoroughly as reasonably possible. This meant we had to decide what to do with people who would have explored only a very limited range of potential uses, applications, or functionalities left to their own devices. With these people, it was necessary to intervene in their natural process of exploration, discovery, and experimentation by at times prompting them to consider uses they did not think of on their own. To manage the issue of researcher influence, we carefully titrated the amount of prompting we gave them, starting with very open-ended questioning and gradually becoming more leading as needed, finally actually proposing a "usage experiment" the person might do for us.

At times, we used an upcoming business activity that we knew was on their agenda to negotiate usage experiments with them. For example, we knew that some people had spontaneously attempted to use the Tablet (which in this version had a detachable keyboard and so could be used in "slate mode") to hand-write responses to e-mails while on the airplane, something they found unpleasant to do in a cramped coach seat using a conventional laptop. If another participant, one who was not as active in trying out the Tablet for new scenarios, was planning a business trip, we could draw on this knowledge. Since we talked casually with participants about their work all the time, it was natural to discuss the upcoming business trip, to talk about the hassles of business trips in general, and to give the person plenty of opportunity to think of some way in which it might be interesting to try using the Tablet to reduce the hassle. This gave us a chance to find out if the idea of using the Tablet on the airplane even occurred to them. If it did not, we could ask progressively leading questions to see if there was anything that it might be useful to try. Finally, we could suggest a specific experiment, like working on their e-mail on the plane, and even negotiate this with them as an assignment.

As stated earlier in the section on methodological challenges of longitudinal research, to make this approach reasonable, it is very important to treat the amount of prompting needed as a piece of data, something that must be taken into account in interpreting the findings. The fact that prompting was needed does not invalidate the data about the user's eventual experience. After all, once a technology becomes widespread, people are exposed to all kinds of suggestions from other users that might induce them to try things they would not have attempted on their own. But for research purposes, one interesting question might be what types of potentially valuable application scenarios are more and less obvious to users.

As mentioned earlier, it is also important to take into account that people who undertake a task on their own might be different from those who undertake it in response to encouragement from the researcher. For example, in this case, perhaps people who had already experienced much success and benefit from writing by hand and converting to text were more likely to decide on their own to try writing e-mails by hand while on the airplane, converting them to text, and then sending them later when they connect to a network. The researcher needs to know enough about each participant to consider any such differences between people who need more prompting and those who need less and take this into account in the interpretation. Furthermore, in the course of "negotiating" usage experiments with the user, there should be an opportunity to explore their expectations of what the experience will be like and to understand their historical experiences that these may be based on.

54.5 WHERE DOES NATURALISTIC USABILITY FIT INTO THE PRODUCT DEVELOPMENT LIFECYCLE?

From the examples given earlier, it may seem as though naturalistic usability requires a fully functional system and, therefore, is applicable only late in the design and development process. Although naturalistic elements can be introduced into even early, rapid usability studies using low fidelity prototypes, the full benefits do require the possibility of a more complete representation of the usage experience. However, if naturalistic usability studies could only happen at the end of the development process, this would certainly be unfortunate, because the understanding they provide incorporates things that should be introduced into the early stages of product planning.

Part of the solution comes from thinking a bit more broadly about what to test. Naturalistic evaluation of competitive products or an earlier version of your own product can be extremely useful in providing guidance for current development efforts. Admittedly, while this is likely to suggest new design directions, it will not directly test them. Therefore, it does not replace iterative testing using simulations. However, it is quite possible to integrate some aspects of rapid prototyping of new interaction design ideas into the naturalistic evaluation. If, through probing of usability difficulties that arise while the user is performing a natural task, the researcher can develop a hypothesis about design changes that may eliminate or reduce the problem, or knows of an alternative design that is under consideration, it is often possible to do a low fidelity test of those alternatives in the context of the user's actual task.

It is certainly easier to produce a functional prototype of some products than of others. Devices that require manufacturing, firmware, and software are expensive to prototype. A device or system that requires a complex installation for realistic use in homes or work places adds another layer of complexity. With these, creation of a functional prototype for testing may be out of the question. On the other hand, a new website design may be fairly easy to implement in a functional version for limited release to test participants. With some software, it is worthwhile and feasible to evaluate in the field in a scaled-down proof of concept version (Allen, McGrenere, and Purves 2008).

Very few products spring forth with no antecedents. In many cases, there is an earlier version of the product in use, naturalistic evaluation of which should feed planning for subsequent releases. Too often, once a product is released, decisions about major changes come primarily from user complaints and requests rather than from disciplined evaluation in the field. Some products have a naturally limited first release that can present an excellent opportunity for naturalistic testing. We once worked on a browser-based tool for enrollment in an innovative employer-provided health insurance program that allowed personalization of a wider range of factors than was typical in health insurance. The initial rollout of this product involved a very limited number of employers, who held an informational fair for employees at the start of their annual open enrollment period. Carrying out usability research during these fairs with employees allowed evaluation with people who were actually signing their families up for coverage.

Finally, naturalistic usability can also be incorporated into other forms of field research, which do tend to happen early in the product development lifecycle or are construed as providing fundamental insights about users. Actual projects in the field often use hybridized methodologies. There is no inherent reason preventing elements of usability evaluation from being incorporated into contextual inquiry studies or even ethnographies. The obstacles may more often have to do with the fact that detailed evaluation of interaction design may not be the primary mission of these other studies, the need to prioritize how you spend your limited time in the field with participants, who the consumers of the research findings are (e.g., product strategists or designers), and the particular skill set of the researcher.

User-centered design includes a very long list of methods and techniques for introducing knowledge of the user into the product planning and design process, and no one technique is enough to ensure that design is fully user-centered. It is important to consider the complementary strengths and weaknesses of different approaches and to choose them consciously rather than out of habits of practice. To whatever degree you can incorporate it into your research program, naturalistic usability evaluation techniques will help bridge the gap between user research and simulation-based usability evaluation.

REFERENCES

Allen, M., J. McGrenere, and B. Purves. 2008. The field evaluation of a mobile digital image communication application designed for people with aphasia. *ACM Trans Access Comput* V1(1), Article 5:1–26.

Barnum, C. 2011. *Usability Testing Essentials: Ready, Set… Test!* Burlington, MA: Morgan Kaufmann.

Beers, R., and P. Whitney. 2006. From ethnographic insight to user-centered design tools. In *Ethnographic Praxis in Industry Conference Proceedings*, 144–54. Arlington, VA: American Anthropological Association.

Beyer, H., and K. Holtzblatt. 1998. *Contextual Design: Defining Customer Centered Systems.* San Francisco, CA: Morgan Kaufmann.

Blomberg, J., and M. Burrell. 2011. An ethnographic approach to design. In *The Human-Computer Interaction Handbook*: *Fundamentals, Evolving Technologies and Emerging Applications*, 3rd ed, ed. J. Jacko. New York: Taylor & Francis Group.

Coursaris, C., and D. Kim. 2007. A research agenda for mobile usability. In *Proceedings CHI 2007*, 2345–50. New York: ACM.

Diggins, T., and P. Tolmie. 2003. The 'adequate' design of ethnographic outputs for practice: Some explorations of the characteristics of design resources. *Pers Ubiquitous Comput* 7(3):147–58.

Dray, S., and D. Siegel. 2009. Understanding users in context: An in-depth introduction to fieldwork in user-centered design. In *Human-Computer Interaction—INTERACT 2009, 12th IFIP TC 13 Conference Proceedings, Part II*, ed. T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. Prates, and M. Winkler, 950–1. New York: Springer.

Dray, S., D. Siegel, E. Feldman, and M. Potenza. 2002. Why do version 1.0 and not release it?: Conducting field trials of the tablet PC. *Interactions* 9(2):11–6.

Duh, H. B., G. C. B. Tan, and V. H. Chen. 2006. Usability evaluation for mobile devices: A comparison of lab and field tests. In *Proceedings of the 8th Conference on Human–Computer Interaction with Mobile Devices and Services—Mobile HCI 2006*, eds. M. Nieminen and M. Röykkee, September 12–15, 181–6. Helsinki, Finland, New York: ACM.

Goodman, J., S. A. Brewster, and P. D. Gray. 2004. Using field experiments to evaluate mobile guides. In *Proceedings HCI in Mobile Guides 2004.* New York: Springer. http://www.dcs.gla.ac.uk/~stephen/papers/MobileGuides04_Goodman.pdf (accessed November 18, 2010).

Hartson, R. H., T. S. Andre, and R. C. Williges. 2003. Criteria for evaluating usability evaluation methods. *Int J Hum Comput Interact* 15(1):145–81.

Holtzblatt, K. 2011. Contextual design. In *The Human–Computer Interaction Handbook*: *Fundamentals, Evolving Technologies and Emerging Applications*, 3rd ed, ed. J. Jacko, Chapter 43. New York: Taylor & Francis Group.

ISO 9241-11. 1998. *Ergonomic Requirements for Office Work with Visual Display Terminals (VDT)s—Part 11 Guidance on Usability*. AM International Organization for Standards.

Jones, R. 2006. Experience models: Where ethnography and design meet. In *Proceedings Ethnographic Praxis in Industry Conference*, 81–93. Arlington, VA: American Anthropological Association.

Kaikkonen, A., T. Kallio, A. Kekäläinen, A. Kankainen, and M. Cankar. 2005. Usability testing of mobile applications: A comparison between lab and field testing. *J Usability Stud* 1(1):4–16.

McDonald, S., K. Monahan, and G. Cockton. 2006. Modified contextual design as a field evaluation method. In *Proceedings of the 4th Nordic Conference on Human–Computer Interaction: Changing Roles,* NordiCHI '06, vol. 189, ed. A. Mørch, K. Morgan, T. Bratteteig, G. Ghosh, and D. Svanaes, 437–40. New York: ACM Press.

Medlock, M. C., D. Wixon, M. McGee, and D. Welsh. 2005. The rapid iterative test and evaluation method: Better products in less time. In *Cost Justifying Usability: An Update for the Internet Age*, ed. G. Bias and D. Mayhew, 489–517. San Francisco, CA: Morgan Kaufmann.

Nielsen, C. M., M. Overgard, M. B. Pedersen, J. Stage, and S. Stenild. 2006. It's worth the hassle! The added value of evaluating the usability of mobile systems in the field. In *Proceedings of the 4th Nordic Conference on Human–Computer Interaction: Changing Roles*, ed. A. Mørch, K. Morgan, T. Bratteteig, G. Ghosh, and D. Svanaes, 272–80. New York: ACM Press.

Salvador, T., G. Bell, and K. Anderson. 1999. Design ethnography. *Des Manag J* 10(4):35–41.

Siegel, D., and S. Dray. 2005. Avoiding the next schism: Ethnography and usability. *Interactions* 12(2):58–61.

Snyder, C. 2003. *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*. San Francisco, CA: Morgan Kaufman.

55 Survey Design and Implementation in HCI

A. Ant Ozok

CONTENTS

55.1 INTRODUCTION

Survey and questionnaire design has been a primary source of data collection within the Human–Computer Interaction (HCI) context since the early days of the science (Baecker et al. 1995). Within the HCI context, surveys are defined as compilations of questions that are implemented either via a computer or paper-and-pencil-based environment, that either have quantitative or qualitative scales, or are open-ended, and that target at extracting a variety of information from a representative sample of the target population (which is in most cases current or prospective users of an HCI system being evaluated).

Survey use is popular in HCI research as it allows researchers to collect, in a relatively easy manner, information based on users' satisfaction, opinions, ideas, and evaluations regarding a system. Design and implementation of surveys are not as costly as conducting experiments in closed environments with special equipment; advances in computer-based survey products and web-based survey services allow direct recording and easy manipulation of survey data by eliminating the need of translation from paper-based to an electronic environment; and, with each survey taking minutes to complete in most cases, given a large sample of potential participants can be reached, surveys are a good resource for collecting large amounts of data in a relatively short amount of time and with minimal resources, especially when compared to controlled objective experimental measures that involve in most cases lengthy tasks and recording sessions. On the other hand, surveys are constantly challenged in terms of their validity and reliability mostly due to their high reliance on participant opinions and the impossibility to measure with full reliability that the questions are answered by participants objectively. Quantitative survey research is also sometimes criticized due to difficulties related to survey scaling, as scales rely on an assumption that participants have the same or similar perceptions of scale responses that are subjective in structure (responses such as "I strongly agree" or "I often do it").

This chapter discusses the different aspects of survey design and implementation in HCI in a structured and comprehensive manner. After a discussion of the purpose and a brief history of surveys, the different types of surveys (content and structure-wise), application domains, design, and evaluation techniques are discussed with illustrative examples. The chapter is concluded with emerging and future trends in the HCI survey design and implementation areas.

55.1.1 Purpose of Survey Use and Survey Applications in HCI

Usability evaluation has been a primary component of HCI since its inception in the 1960s. User performance and satisfaction have long been tapped as the major components of usability testing and evaluation (Shneiderman 1992; Nielsen 1989). While user-performance measurement relies on objective methods such as software-based time and error measurement, user satisfaction requires more sophisticated tools to be measured objectively. User satisfaction is defined as the level to which a system meets its users' standards and requirements (Hackman and Oldham 1980).

Directly relating to user satisfaction, user preferences, opinions, and evaluations concerning HCI systems are also of strong interest to usability testing and evaluation processes. Since it is not possible to measure all of these usability components through unequivocal measurement methods, explicit tools have been developed to elicit information regarding user satisfaction, preferences, opinions, and evaluations both qualitatively and quantitatively through user surveys. Surveys serve this specific purpose well by posing targeted questions to users for the purposes of HCI design and evaluation.

While surveys can be designed to collect a variety of types of information concerning the target population, relevant to HCI research and literature (Card 1996), they are mostly targeted at collecting information in the following three categories:

1. User Evaluation: The category aims at collecting information regarding how much a system, product, or environment meets user goals, expectations, and standards. In this category, users are asked a number of questions regarding whether their overall impression regarding the object being evaluated is high, what exactly constitutes this impression, what and where exactly the problems are, and so on. Relating to user satisfaction, this category is also about determining user opinions specific to products or systems, where questions can also include users' opinions concerning whether tasks can be completed effectively and efficiently, whether the system is fast, and so forth.
2. User Opinion: The category can, but does not have to, be specific to products, systems, or environments. These types of surveys are aimed at determining what users think about the requirements from a system, product, or environment to fulfill its function satisfactorily. Examples can include surveys that aim at needs assessments for the next generation of cell phones (e.g., what new functionalities can be useful in newer cell phones besides those that already exist according to cell phone customers). Simply put, while the former category consists of surveys regarding the performance of existing systems, environments, and products, the current category is concerned with what users think about what might be useful concerning these systems in more general terms.
3. Others: The third category includes the remaining possible survey types aimed at collecting a number of different information types within the HCI context. One such category consists of surveys that are strictly concentrated on population demographics. These types of surveys do not contain questions relying on participants' evaluation of specific products or their opinions, but rather solely on qualifications they own, such as age, sex, education level, skill level, and so forth, or things they do, such as how frequently they go on the Internet or use a cell phone. These types of survey questions are less based on opinion-heavy responses than the previous two categories.

55.1.2 Brief History of Survey Design in HCI

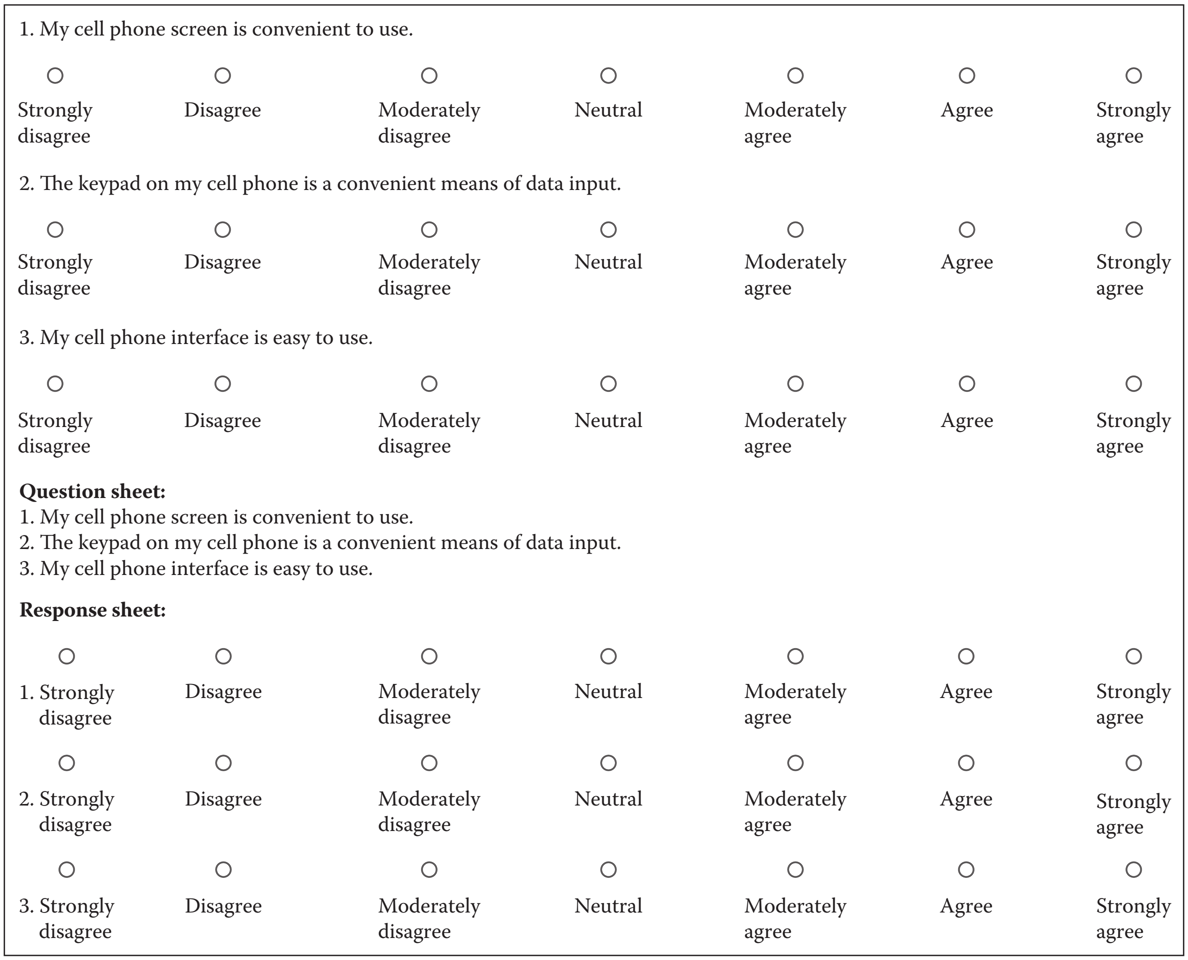
Surveys started being used as a computer science and, to a limited extent, an HCI research tool in early 1970s, borrowing techniques from anthropology and experimental and social psychology (Myers, Hollan, and Cruz 1996). With contributions from developments in the overall survey administration and language issues (Converse and Presser 1986; Belson 1981; Jenkins and Dillman 1997), researchers discovered early on that information regarding user attitudes, preferences, and evaluations within the context of computer technology (software and hardware) development can be collected fairly easily with paper-and-pencil surveys. Hence, in the 1970s and 1980s, user surveys were a part of computer research with a social psychology emphasis, but not directly relating to usability testing and usability-design issues. In the mid-1980s graphical user interfaces became an important part of end user computing and usability research took off. The surveys gained a more significant role in HCI research around the same time, and with the advent of graphical user interfaces, surveys in HCI and specifically usability research gradually gained importance. With the graphical user interfaces as we know today gaining high popularity with Windows 95, usability research accelerated (Myers 1996), and besides building usability laboratories, companies and research institutions started developing and implementing surveys to determine user trends and preferences in HCI. Advanced electronic and paper-and-pencil survey-design methods have been developed in the last decade (Dillman 2000), and user surveys have become an essential part of data collection in HCI research.

55.1.3 Paper-and-Pencil and Electronic Surveys

Survey implementation largely relies on practical factors. Besides the challenge of finding a sample size that is both large enough and representative of the population the study is targeted at, implementation challenges include presenting the participant sample with a survey that is quick and easy to complete, has a fun factor and a learning component for the participants, and does not require tedious data extraction and manipulation once implementation is completed. Additionally, surveys in every topic should be unambiguous, unbiased, correctly coded, piloted, and ethical (Stone 1993). Today's surveys are almost universally implemented in two forms: (1) paper-and pencil surveys, which require participants to mark or write their responses on response sheets that also contain questions, either on the same sheet or separately; and (2) electronic surveys, which require the users to use the screen, keyboard, and/or mouse of a computer to mark or type their responses on the screen.

Paper-and-pencil-based surveys require the survey material to be physically distributed, filled out, and returned. This process can occasionally be cumbersome and tedious. Moreover, these types of surveys also require manual entry of quantitative, and in most cases qualitative, data for analysis. One solution to the problem of translation of paper-based data into electronic format is using Scantron sheets, which are sometimes referred to as "bubble-sheets." In this system, designated areas (bubbles) on printed sheets are filled out by participants with a pencil, and these sheets are then fed into a computer and scanned for correct answers. This process, however, is quite costly due to the scanning equipment necessary for the process. Figure 55.1 shows two sample paper-and-pencil survey sheets, one with response spaces below each question and one with response sheets that are separate from the question sheets.

Although about 62% of all American households own one or more computers (U.S. Census Bureau 2005), computers are still not accessible to the entire population. Therefore, paper-and-pencil surveys are still widely popular. Paper-and-pencil surveys allow swift collection of large data quantities if they are administered to groups of participants simultaneously, such as a group of students during a class period who are asked to fill out and return the surveys immediately. One



1. My cell phone screen is convenient to use.

○	○	○	○	○	○	○
Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree

2. The keypad on my cell phone is a convenient means of data input.

○	○	○	○	○	○	○
Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree

3. My cell phone interface is easy to use.

○	○	○	○	○	○	○
Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree

Question sheet:
1. My cell phone screen is convenient to use.
2. The keypad on my cell phone is a convenient means of data input.
3. My cell phone interface is easy to use.

Response sheet:

○	○	○	○	○	○	○
1. Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree
○	○	○	○	○	○	○
2. Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree
○	○	○	○	○	○	○
3. Strongly disagree	Disagree	Moderately disagree	Neutral	Moderately agree	Agree	Strongly agree

FIGURE 55.1 Two paper-and-pencil presentations of the same survey, one with the response spaces below each question and one with separate question and response sheets.

other common way of paper-and-pencil-based survey implementation is mailing surveys to participants and asking them to return them via mail, in most cases in postage-prepaid envelopes. However, recent studies indicated that return rates of mailed-in surveys by the participants is highly variable depending on the type of survey (Dillman 1991, 2000). Taking into consideration the percentage rate of computer ownership among American households, mailed-in surveys can be concluded as a less-preferred means of data collection specifically within the context of the HCI research, and mail surveys are therefore not popularly used for HCI research purposes.

Computer-based surveys (sometimes referred to as PC-based surveys) have become popular with the advent of home computers in the 1990s, even before the high adoption of the World Wide Web. In computer-based surveys, participants are presented the survey on a specific, standalone (nonnetworked) computer. It should be noted that these types of surveys are not web-based, but rely on the software installed on the computer on which the survey is implemented.

Participants use the mouse to click on their responses of choice for multiple-choice questions. Mouse actions are generally implemented on dropdown combo boxes (where a dropdown menu opens up with the options when the user clicks on the button located to the right side of it) or radio buttons (a round area to the left of the option is clickable), or check boxes where multiple choices can be clicked on one at a time (a square-shaped box to the left of the option is clickable) (Ozok and Salvendy 2000). For text entries, participants can type text on specified text boxes. While computer-based surveys can be convenient because of having the initial data in electronic format and eliminating the necessary transformation to electronic format in paper-and-pencil-based surveys, they require the participants to be stationary on a specific computer. For large-size implementations, computer-based surveys can be extremely slow in collecting the data, mainly due to limited computer equipment and scheduling difficulties. It can be concluded that while computer-based surveys can be advantageous in the data-analysis stage and are still popular in kiosks stationed in public places such as shopping malls, they are not suitable for large-sample size and lengthy surveys, and hence are not the best solution in survey-based data collection in on HCI context.

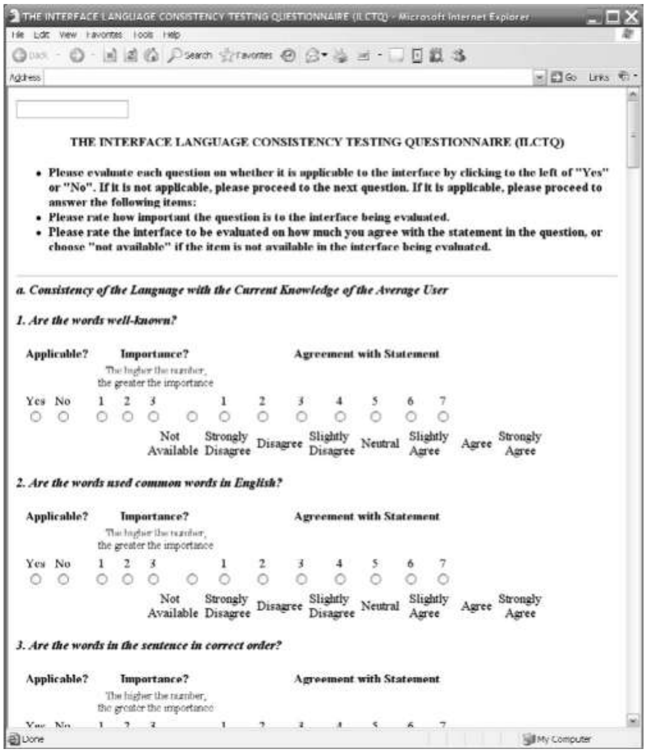
With the advent of the Internet, web-based (online) surveys have become highly popular (Dillman 1999) and allow researchers to conduct surveys more effectively and efficiently than more traditional means (Zhang 2000). Server-based software allows survey participants to access a web page and fill out the survey, then submit their results mostly to a central server by clicking an on-screen button. Web-based survey interfaces in structure look very similar to computer-based surveys with the same interface elements of dropdown combo boxes, radio buttons, check boxes, and text boxes serving the functions of various data-entry types by the participants. The data are collected on a central web server in these types of surveys, which can be easily obtained and manipulated by the survey administrators. Additional data storage and analysis programs residing on these web servers can compile the data in a variety of formats such as Microsoft Excel, and also implement some automatic data analyses such as calculation of means and standard deviations (descriptive statistics).

One additional electronic survey type consists of the administration of e-mail surveys in which participants are e-mailed a survey and are asked to fill it out and e-mail it back. However, with the dramatic increase in the amount of spam e-mail users receive in recent years, it can be concluded that these kinds of e-mails are likely to be perceived as spam and are likely to be discarded. Therefore, e-mail surveys are not articulately covered in this chapter.

There are two types of methods used in web-based survey administration today. One method is to use a web survey administration service provider (such as SurveyMonkey.com) by paying it a monthly or yearly fee. In most cases, various packages are available ranging from a small number of surveys to unlimited administration of surveys. The web service providers also have a variety of options for the retrieval of survey data by the administrators, for example, in Access or Excel formats, with some, in most cases basic, statistical analyses (such as descriptive statistics and correlations) already performed on the data. Additionally, the services also give flexibility in customization of survey interfaces such as giving the freedom to the administrators (and in some rare cases to the participants) to choose their text and background colors, font sizes and types, how many questions to have per web page, and so on. Today, there are more than a hundred online survey-service providers with monthly fees varying from $3 for small-scale academic surveys to $500 for large-scale, mostly marketing-related surveys.

Another means of administering online surveys is to use one's own hardware and software. As a central server is necessary for collection and storage of the data, this hardware can either be purchased or rented, or an existing server can be used. The amount of storage space necessary largely depends on the length and sample size of the survey, but since most surveys contain textual data, it is almost unimaginable to need more than five gigabytes of storage space for a large-size survey for purposes of HCI research. A variety of open source (such as php ESP [Easy Survey Package]) and licensed software vendors (such as Inquisite) are available for survey implementation and administration on administrator-owned servers. Like survey-service providers, these software packages also allow a variety of customization flexibilities concerning the survey interfaces. Figure 55.2 presents a sample interface from an online survey.

For both paper-and-pencil-based and electronic surveys, human cognitive limitations should be taken into consideration and basic human factors guidelines should apply. Paper-and-pencil surveys should be administered with pencil to allow participants to go back and change their responses if they want to. White paper with black, 12–14-point-sized Times New Roman font text should be used, as those font sizes are the most common and easily readable text sizes in printed documents. Labels should be presented with bold,



THE INTERFACE LANGUAGE CONSISTENCY TESTING QUESTIONNAIRE (ILCTQ) - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address
Go Links
Done
My Computer

THE INTERFACE LANGUAGE CONSISTENCY TESTING QUESTIONNAIRE (ILCTQ)

- Please evaluate each question on whether it is applicable to the interface by clicking to the left of "Yes" or "No". If it is not applicable, please proceed to the next question. If it is applicable, please proceed to answer the following items:
- Please rate how important the question is to the interface being evaluated.
- Please rate the interface to be evaluated on how much you agree with the statement in the question, or choose "not available" if the item is not available in the interface being evaluated.

a. Consistency of the Language with the Current Knowledge of the Average User

1. Are the words well-known?

Applicable? Importance? Agreement with Statement
The higher the number, the greater the importance
Yes ○ No ○ 1 ○ 2 ○ 3 ○ Not Available ○ 1 ○ Strongly Disagree 2 ○ Disagree 3 ○ Slightly Disagree 4 ○ Neutral 5 ○ Slightly Agree 6 ○ Agree 7 ○ Strongly Agree

2. Are the words used common words in English?

Applicable? Importance? Agreement with Statement
The higher the number, the greater the importance
Yes ○ No ○ 1 ○ 2 ○ 3 ○ Not Available ○ 1 ○ Strongly Disagree 2 ○ Disagree 3 ○ Slightly Disagree 4 ○ Neutral 5 ○ Slightly Agree 6 ○ Agree 7 ○ Strongly Agree

3. Are the words in the sentence in correct order?

Applicable? Importance? Agreement with Statement
The higher the number, the greater the importance
Yes No 1 2 3 1 2 3 4 5 6 7

FIGURE 55.2 Sample interface from an online survey.

16–18-point-size text, and while pictures are seldom presented to participants, if they are presented, they should have enough resolution to appear clearly on paper. In short, paper-and-pencil-based survey interfaces should be inspected to make sure they comply with the structural human-factors guidelines for paper-based interfaces.

Similarly, electronically administered surveys (web- or PC-based) also should follow basic human-factors guidelines for computer interfaces. For web-based surveys, it should be noted that participants will access the surveys from a variety of different types of machines, and basic computer and web design guidelines indicate 800 × 600 color screen resolution is the most common screen type (Nielsen 1993; Badre 2002) which should be taken into consideration when survey interfaces are designed, making sure that survey objects, mainly text fonts and some images if there are any, are easily visible on screens with this resolution. The basic rule of black text on white background in web usability should also be applied in computer- and web-based interfaces, and screen objects such as dropdown combo boxes, radio buttons, buttons, and check boxes should be the same size (mainly height) as text lines to ensure consistency and easy visibility and to allow users to click on them easily. Text boxes for the users to type in should allow text size consistent with the text on the question parts of the interface. Overall, again, it can be said that in web-based surveys, basic web-design guidelines can easily be adopted.

Additionally, one important item in survey design is to prevent survey participants from getting worried about the excessive length of the surveys, as too-long surveys may result in significantly decreased response rates (Lund and Gram 1998; Krosnick 1999). Taking also into consideration the fact that computer users don't enjoy scrolling down the screen in general (Shneiderman 1992), no more than 20 questions should be presented on one survey screen. For surveys containing more than 20 questions, subsequent questions should be presented on subsequent screens (again, each screen not exceeding 20 questions), which participants should move to by clicking on a screen link or button that should have a statement such as "Click here to continue."

Both paper-and-pencil-based and electronic surveys will continue to be used in HCI research. While it is unlikely that the former will go extinct any time soon, recent studies (such as Dillman 2000) showed that web-based surveys have very significant advantages in data collection and analysis stages of survey-based research. The fact that most HCI-related survey research uses computer-savvy sample participant groups is also a factor that helps the popularity of web-based surveys within HCI. It is therefore expected that web surveys will eventually constitute a large majority of survey-based research in HCI. While no previous study explicitly investigated the exact share percentage of paper-and-pencil-based and electronic surveys in HCI research, it is estimated that more than 60% of all surveys for HCI research are implemented on the web environment.

While the discussion in this chapter mostly concentrates on structural issues of paper-and-pencil-based and electronic survey design, a much larger research topic concentrates on how the actual survey content should be designed, created, and implemented. The next chapter discusses content creation in surveys within the context of HCI research.

55.2 SURVEY DESIGN IN HCI

A major part of survey-design research in HCI is concerned with the formulation of survey questions and scales. In comprehensive user surveys, questions may be presented as one large, continuous list. Related questions may also be presented in categories and sections, such as the demographic questions in the first section, questions evaluating the structural elements of the interface in the second section, and so on. As discussed earlier, surveys in HCI research are mostly concentrated on collecting two categories of information: participants' opinions and evaluations. These two survey types are discussed in the next section. This discussion is followed by an explanation of survey application domains and the survey design methodology.

55.2.1 Survey Types

55.2.1.1 Opinion Surveys

Opinion surveys aim at determining what participants think about issues pertaining to certain concepts. Additionally, opinion surveys can also measure participants' wishes, habits, and customs (Baker 1998). The way opinion surveys differ from evaluation surveys is that opinion surveys are not centered on a specific product or environment, but are more general in nature. For example, an opinion survey can target measuring cell phone users' habits and opinions regarding the use of the cell phone interface (whether they are happy with the screen size, whether they can easily enter data using the keypad, etc.). While questions are focused on a product in this example (e.g., a cell phone) the population of cell phone users is highly diverse, as are cell phone devices that are used by the population. Hence, such an opinion survey will measure general trends in usability concerning cell phone interfaces rather than evaluating a specific cell phone interface.

Opinion surveys are administered with more freedom than evaluation surveys, as the participants do not have to have the evaluated issue, product, or environment fresh in their memory. Hence, they can fill out opinion surveys at any time and in any environment that is convenient for them. Opinion surveys include questions that do not require any recalling process, meaning they contain questions which participants can answer without having to recall a specific feature or part of the environment, product, or issue. As opinion surveys ask general questions about participants' current standing regarding opinions, wishes, customs, and habits, the answers may sound subjective and vary greatly among participants. Sample questions for opinion surveys include statements such as "Does the use of computers in daily tasks contribute to your overall technology knowledge?" or "Are you in general satisfied with the amount of product information available on an e-commerce page you frequently visit?" or "How would you rate the customer services on the sites where you frequently shop?"

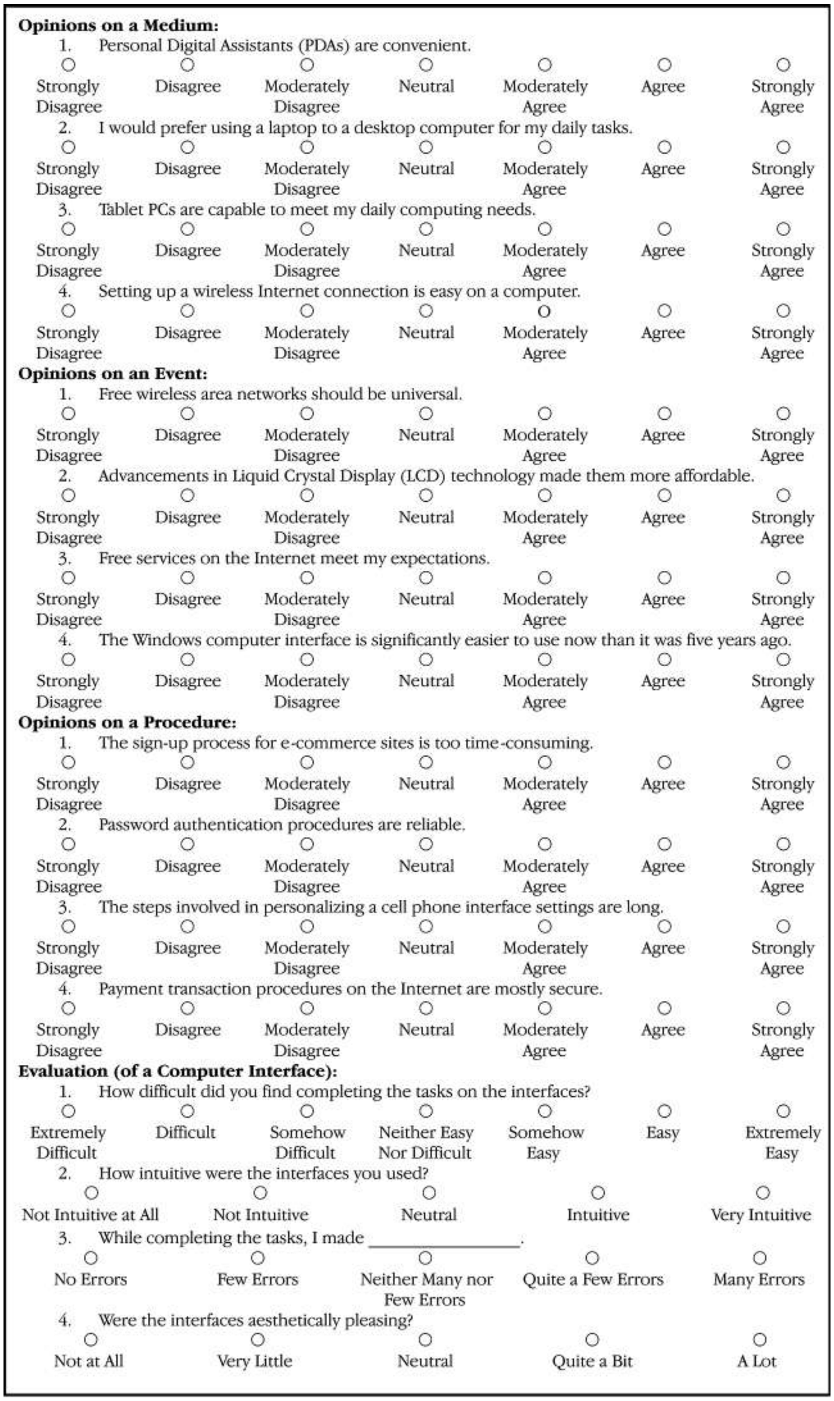
Opinion surveys can cover a broader variety of issues than evaluation surveys, which are more focused. They can include both qualitative and quantitative scales for their responses. Although no previous literature came up with a strict classification of opinion surveys, the following classification can help in structuring of opinion surveys and what kind of questions should be asked for what types of opinion-related survey categories in HCI research (sample questions for each classification of opinions are presented on Figure 55.3):

a. Opinions on a medium: Within the context of HCI research, these types of surveys concentrate in most cases on interface design, usability, user satisfaction, and user preferences issues concerning a medium the participants use in their daily lives. Most popular examples of this type of media include daily-used devices such as computers, Personal Digital Assistants (PDAs), and cell phones. These types of surveys concerning users' opinions on a certain medium aim at determining general trends in user opinions concerning whether the design of the medium is satisfactory and meets user needs and requirements, whether there are problems concerning the medium, and what can be possible solutions. Opinion surveys concerning a medium are also useful when they are used by usability specialists and engineers to develop new versions of products or to come up with new products, as the survey results can pinpoint the needs of the target population to be met regarding the medium. Sample questions for these types of surveys can include questions like, "Are you in general satisfied with the web browsing capabilities of your cell phone?" or "What additional capabilities would you like to see on your PDA user interface?"
b. Opinions on an event: Within the context of HCI research, user opinions on an HCI-related event can include what they think about certain HCI-related happenings. Examples can include opinions concerning certain HCI-related activities, with questions like, "Do you find the move from a text-based interface to a graphical user interface helpful?" These types of surveys are rarer in nature and aim at collecting basic trends concerning user opinions in current HCI-related activities.
c. Opinions on a procedure: These kinds of surveys aim at determining the user opinions on procedures to complete HCI-related tasks. They are similar to medium-related opinion surveys, but rather than questions about the medium itself, these surveys have the goal of determining user opinions on how it is used. In web and e-commerce design, these kinds of opinion surveys are helpful in determining whether the procedures to complete general tasks (for example, web navigation) meet user requirements and needs. A sample question in an opinion survey concerning a procedure could be, "Are you satisfied with how long it generally takes to purchase a product on an e-commerce site?" Surveys to explore opinions on a procedure are less common as HCI researchers usually resort to evaluation surveys to test procedures for use of computer interfaces and other media (explained in the next section).

55.2.1.2 Evaluation Surveys

More specific than opinion surveys, evaluation surveys (or questionnaires) are generally administered after a certain procedure is implemented on the participant group. While opinion surveys can be administered at any time to the participants, evaluation surveys are administered right after the participants have completed a certain procedure. In evaluation surveys, participants are asked about tasks they have just recently completed. Therefore, evaluation surveys are in most cases preceded by certain usability-related experimental procedures. Most common in HCI, evaluation surveys are administered after participants have completed a number of tasks in controlled computer environments. They are also implemented right after the procedure to ensure that memories regarding the procedure are still fresh in the participants' minds, as evaluation surveys require a high amount of recall of procedures and interfaces in the tasks that were just previously completed.

HCI-related evaluation surveys have the main goal of evaluating usability, user satisfaction and user preference issues concerning user interfaces or environments (Ozok and



Opinions on a Medium:

1. Personal Digital Assistants (PDAs) are convenient.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
2. I would prefer using a laptop to a desktop computer for my daily tasks.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
3. Tablet PCs are capable to meet my daily computing needs.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
4. Setting up a wireless Internet connection is easy on a computer.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

Opinions on an Event:

1. Free wireless area networks should be universal.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
2. Advancements in Liquid Crystal Display (LCD) technology made them more affordable.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
3. Free services on the Internet meet my expectations.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
4. The Windows computer interface is significantly easier to use now than it was five years ago.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

Opinions on a Procedure:

1. The sign-up process for e-commerce sites is too time-consuming.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
2. Password authentication procedures are reliable.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
3. The steps involved in personalizing a cell phone interface settings are long.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree
4. Payment transaction procedures on the Internet are mostly secure.
   ○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

Evaluation (of a Computer Interface):

1. How difficult did you find completing the tasks on the interfaces?
   ○ Extremely Difficult ○ Difficult ○ Somehow Difficult ○ Neither Easy Nor Difficult ○ Somehow Easy ○ Easy ○ Extremely Easy
2. How intuitive were the interfaces you used?
   ○ Not Intuitive at All ○ Not Intuitive ○ Neutral ○ Intuitive ○ Very Intuitive
3. While completing the tasks, I made _______________.
   ○ No Errors ○ Few Errors ○ Neither Many nor Few Errors ○ Quite a Few Errors ○ Many Errors
4. Were the interfaces aesthetically pleasing?
   ○ Not at All ○ Very Little ○ Neutral ○ Quite a Bit ○ A Lot

FIGURE 55.3 Opinion and evaluation survey sample questions.

Salvendy 2001). After certain tasks are completed in these controlled environments, evaluation surveys aim at determining the exact nature and location of problems and points open to improvement in the human–computer environments. Therefore, evaluation surveys are in most cases relatively detailed in nature. In a sample procedure, for example, participants can be presented a number of web pages and asked to complete some common user tasks on those pages, tasks that can include text entry, direct manipulation, and form filling. An evaluation survey that would follow could include questions such as "Was the site navigation difficult?", "Was the text size on the site readable?", "Did you enjoy the overall experience on this site?", "Would you visit this site again?", and so on.

As their name indicates, evaluation surveys aim at evaluating interfaces, environments, and procedures from the user's perspective. For that purpose, they are tools to determine how participants evaluated those interfaces, environments, and procedures. In that sense, evaluation surveys are explicit and not that much different from objective measurement methods such as performance measurement in HCI tasks. Similar to opinion surveys, evaluation surveys use both qualitative and quantitative scales.

Evaluation surveys are not only helpful in evaluation of product interfaces, environments, and procedures. They can also be used in evaluating certain conceptual features. For example, a researcher may be investigating whether the introduction of a certain interface concept results in higher user satisfaction regarding the interface. If this particular feature is, say, that of interface visibility, the survey administrator can first present the participants with high-visibility and low-visibility computer interfaces, and the subsequent evaluation survey can contain questions such as: “Did higher visibility improve your overall satisfaction with the computer screens?”

Evaluation surveys are useful in a variety of HCI-related commercial and academic research activities. In academic research, different computer or computer-related interfaces can be evaluated through surveys to determine whether they result in better user preferences or higher user satisfaction. In commercial research, newly developed products (for example, a new computer peripheral), environments (for example, a new computer interface), or procedures (for example, the steps it takes to complete a transaction with a new e-commerce site design) can be empirically evaluated by having participants complete tasks with those products and procedures or in those environments, then filling out evaluation surveys consisting of detailed questions regarding their satisfaction and preferences regarding the said product, environment, or procedure. While quantitative evaluation results can give statistical backing to user evaluations, helping boost their conclusiveness, qualitative evaluation results can give the researchers who administered the surveys new ideas to improve usability/user-preferences-related design components. Therefore, evaluation surveys are the most commonly used survey types in HCI research and constitute one of the most common and effective user evaluation methods in HCI in general. Examples of opinion survey questions are also presented in Figure 55.3.

55.2.1.3 Other Survey Types

Besides the two main survey types mentioned earlier, one widely used survey type is the demographic survey. Although demographic surveys are almost universal, it would be incorrect to categorize them at the same level as opinion and evaluation surveys, as most user surveys have a section concerning user demographics. Hence, in most cases demographic surveys are essential parts of opinion and evaluation surveys rather than stand-alone surveys. In some cases HCI researchers administer surveys consisting of demographic questions only—for example, to determine the demographics of a user group using a specific application. However, more commonly, HCI surveys consist of opinion and/or evaluation questions in addition to the demographic questions.

Demographic questions play an important part in HCI-related survey design, as most variables of interest in HCI research are also dependent on the factors that are specific to the target population. Consequently, research findings can only be generalized in most cases to the target population from which a representative survey is sampled.

Demographic surveys (or survey sections) in most cases consist of a standard set of categories: age, sex, education, and occupation of the participant. Age can also be asked in the form of “birth year,” and for the question regarding sex, options in the form of “Male/Female” can be presented to participants to mark on the computer or with the pen/pencil. Education level can be left to type in or write, or options can be presented. The question is usually formulated as: “What is the highest degree for which you won a diploma?” or “What is the highest degree you earned?” Typical options for this question are “Elementary school,” “Middle school,” “High school,” “College or university,” “Graduate degree,” and “Post-doctoral graduate degree.” The occupation question is about what kind of job the participant has. For this question, usually the participants are asked to type or write in the designated area, due to the high variety of possible occupations participants may have, although presenting options for this question is also possible if some general occupation categories are all that is needed, for example, options such as “Private sector,” “Academia,” “Student,” and so on.

In addition to this basic set of survey questions, demographic surveys can also include general questions regarding daily habits or current standing issues concerning the participants. Most commonly, demographic surveys in the HCI area contain questions regarding computer use habits of participants, such as “How many times a day do you check your e-mail?” or “How many times in the last year did you shop from a web-based e-commerce company?” These types of questions are usually customized according to the type of information needed for the specific research being conducted and can therefore greatly vary in nature, but in principle they aim at collecting information on computing-related habits in most cases. Table 55.1 presents a set of sample demographic questions as part of a survey regarding cell phone use.

It should be noted that demographic questions are of a more personal nature than opinion and evaluation questions. Some participants may feel that their privacy rights are being violated by being asked to provide their age and education level. For this reason, providing anonymity in surveys and informing the participants about their provided information being not personally identifiable—in other words, providing anonymity—is greatly crucial in HCI research. Knowing that their data cannot personally identify them usually takes care of privacy worries and is known to improve participant enthusiasm. Challenges concerning privacy in survey implementation are described later in this chapter.

There are no other significant survey types widely used in HCI research. Some niche survey types may still exist but are

TABLE 55.1
Sample Demographic Questions from a Survey on Cell Phone Use

Your Age:
Your Gender:
Your Occupation:
How many times a week do you go on the web?:
_____ Less than once a week
_____ Between once and three times
_____ Between three times a week and every day
_____ Every day

In the past year, how many times did you shop online (Please put a number)?: ____

Do you own a cell phone, a Personal Digital Assistant, or a Combination Device?
_____ Yes
_____ No

In the past one year, how many times did you shop online using a cell phone, a Personal Digital Assistant, or a Combination Device (Please put a number)?: _____

few and far between. Therefore, surveys aimed at collecting information relevant to HCI research usually belong to one of the categories explained in this section. In the next section, application domains of surveys relating to HCI research are discussed.

55.2.2 Survey Application Domains

Survey applications are highly popular in a broad range of application domains, in areas ranging from social sciences to marketing, education to production, customer to worker satisfaction, and many more. Today, results obtained from surveys, which in most cases ask comprehensive questions, are deemed reliable and valid in both scientific and industrial projects. The most common application domains of survey research include the following:

- Sales and Marketing: Companies that offer products and services for both consumers and industries use customer/client surveys for both needs-assessment and evaluation purposes. A large number of companies are also solely dedicated to implement customer surveys for companies, analyze the data, and deduct conclusions for sales and marketing purposes. Customer satisfaction, product evaluation, customer relationship management, and customer demographics are only a few of the topics surveyed by sales and marketing forces.
- Medicine: Medical research is not limited to trials relying on objective measurements. Surveys can be helpful in collecting patient data for development of medicine or treatments.
- Education: Educational surveys can help determine population preferences in education as well as education levels and education-related difficulties among population segments.
- Information Technology Research: In the field of information technology, surveys are widely used in connection to software and hardware design and evaluation, covering a broad variety of areas including software engineering, systems analysis and design, and of course HCI, which this chapter covers.

The earlier-mentioned relevant list covering the application domains of surveys is far from complete, but a sample of application domains are presented in the list. The list will no doubt continue growing with the advent of new technologies and sciences. The HCI area is seen as a major application domain of surveys, and is expected to continue to be so.

55.2.3 Survey Design Methodology

Survey design is a methodological activity that requires a systematic design procedure. Survey design techniques and procedures are discussed in this section, including content design and scale design, followed by survey design and redesign issues, a survey design example, and a discussion of challenges in survey design.

55.2.3.1 Survey Design Techniques

Survey design mainly consists of two components: the design of the survey content and the survey scale. They are both discussed in this section.

55.2.3.1.1 Content Design

In the heart of the survey research lays the issue of producing the actual questions to ask the participants. Designing the survey content is actually producing these questions along with their scales. Deciding on which questions to ask the participants in the survey largely depends on three resources: literature, expert opinions, and individual experiences.

A large number of survey questions are based on previous research in the focus area. Relying to some extent on previous literature allows the researchers to achieve high validity of their survey structure, as previously validated research allows current survey design to have strong backing in terms of its content and the targeting of the questions concerning the particular research topic. Therefore, it is best to have backing from previous studies at least for the majority of the questions while designing the survey.

While a designed survey's content may consist largely of questions that are based on the relevant literature in the area, there will be most likely some issues that are intended to be included in the survey but are not covered in the previous literature. Therefore, HCI researchers sometimes rely on experts in the area to cover additional points to be included in the survey. A preliminary survey may be sent in this context to the area specialists to determine the most significant items to be covered in the investigated area. For example, if a survey research is trying to determine the most significant interface design items in e-commerce that affect buying behavior, a preliminary survey may be sent to experts in the area (for example, e-commerce company managers and professors in business schools specializing in e-commerce) to determine the general classifications of interface issues relating to the buying decision. In addition to the literature, these responses from experts can be used as a major resource of question generation for the resulting survey.

Researchers can also rely on their own heuristics and expertise in producing questions. To prevent being accused of "making up the questions," the researchers would need to explain logically why the questions were included in the survey. In these cases, researchers can include questions based on what they think is a significant part of the research item being investigated, or based on the impression that, although the literature did not explicitly point out the issues in these types of questions, there was an implicit indication in previous research towards this particular direction.

Design of survey content is not a difficult task once the researcher has a reasonable background in the area of interest. One common mistake done in design of survey content is the researchers missing important questions during the design and ultimately not addressing those questions. Therefore, cautious, repeated reviews and revisions are necessary before the final implementation of the survey.

55.2.3.1.2 Scales and Open-Ended Questions

Just as important as the content of the questions, the scales for the survey questions in HCI are essential for the accuracy and validity of survey results. Scales are created to attribute numerical values to participant responses in the survey, thereby allowing statistical analyses and giving statistical backing to conclusions obtained from the research. To respond to a scaled question, the participant marks one of the several options, the option which best represents his or her opinion regarding the item in the question. If the question has a large variety of possible answers, or if it requires a lengthy answer, then an open-ended response style may be preferred rather than presenting a scale to the participant. For open-ended responses, participants are mostly given the freedom to write or type as much as they would like.

Both scaled and open-ended questions are suitable for different question types and types of information being obtained from the participants in the survey. Quantitative studies have to use numerical scales to statistically test their hypotheses and support their findings. Qualitative research, on the other hand, analyzes survey data without the involvement of numbers. Because participants have a much higher degree of freedom when responding to open-ended questions, qualitative responses are not restricted to the defined response universe determined by survey designers (also referred to as "survey authors"). On the other hand, conclusions derived from qualitative responses may be more arguable because they cannot be tested statistically.

One more type of response in HCI surveys includes participants being given the freedom to mark more than one response choice. While these types of responses are generally not assigned numerical scales, these types of responses are presented in demographic surveys. In these types of questions with possible multiple responses and in open-ended questions, it is useful to present the option of "Other (please specify):" to the participants, as there is always a possibility that the survey designers may not present the option which the participant would like to give as a response. A sample question of this sort could be "Where do you generally access the Internet?" with the possible responses "Home," "Work," "School," "Coffee Shop," "Internet Café," and "Other (please specify)."

It should be noted that one alternative to open-ended survey questions are interviews and focus groups, and these more interactive data-gathering techniques are likely to result in the collection of richer data than open-ended survey questions as they allow real-time interactions between the researchers and participants. Therefore, it is not highly common in HCI research to use surveys with open-ended questions only. In most cases, especially in quantitative survey research, a mix of both open-ended and scaled questions often proves to provide the best empirical results. Due to their higher frequency of use, this book chapter is more focused on the design and implementation of scaled surveys rather than surveys with open-ended questions. As part of this direction, scale design is discussed in the next section.

55.2.3.1.3 Scale Design

While a large variety of scaling techniques are available for surveys in sociology and psychology research, HCI surveys mostly rely on Likert scales (Medsker and Campion 1997). While contrast scales consisting of yes-or-no questions with 1/0 corresponding scales are also used, five- and seven-point Likert scales are highly common (Aiken and Lewis 1996). In most cases, a scale needs to consist of an odd-number of options. This way, the option in the middle can correspond to a "no preference" or neutral" opinion (Dillman 2000). Each response on the scale is attributed a number to allow the researchers to conduct statistical analysis on the collected data. Item scales need to be kept consistent during the data analysis phase, meaning items should be lined up in the same direction, whether they are positive or negative—in most cases positive responses scoring high and negative responses scoring low on the scales. Inverted questions (questions that ask items in the opposite direction, as discussed later in this chapter) should have their scales reversed in order to keep consistency and allow correct data analysis. A large amount of data analysis mistakes in surveys usually happen because of scaling problems. For example, if the researchers forget to invert scales of reverse questions, then correlations and differences between responses will not come out correctly, resulting in lack of validity of research conclusions.

Scales can indicate a number of different issues. Some scales are concerned with user opinions while others are concerned about frequencies. Most common scale types include agreement measurement ranging between "Strongly agree" and “Strongly disagree”; frequency measurement ranging between “Not at all” and “Very often”; quality opinions ranging between “Very good” and “Very poor”; opinions regarding probability ranging between “Very unlikely” and “Very likely”; and so on. It should be noted that survey scales offer a certain amount of freedom to survey designers on how to name the possible response options for their questions, and therefore scales come in many different varieties, from those measuring amounts (a lot, quite a bit, etc.) to frequencies (very often, often, etc.) to yes-or-no scales. Normally, in most cases a “Not Applicable” or “Not Available” option needs to be added to the scale. When this option is marked, this particular question of this particular subject is eliminated from the analysis. Note that this is different from giving a score of zero to that question in the analysis. Table 55.2 presents a sample of possible response scale sets along with possible number correspondences to the responses.

55.2.3.2 Survey Evaluation Techniques

After the initial design of the survey questions, scales, and instructions to the survey participants on how to fill the surveys out, surveys need to be evaluated to determine whether they are measuring what the designers intended them to measure, whether they are reliable, and whether they produce valid results. Pilot testing is one common method to preliminarily address these issues. However, full evaluation of a survey can mostly happen only after substantial data have been collected with the survey as a tool. Therefore, the evaluation of a survey is conducted based on the data collected by it, and the two components of survey evaluation are the measurement of the survey’s validity and reliability.

TABLE 55.2
Possible Survey Responses and Their Numerical Equivalences

Strongly Disagree	1	***Never***	1
Disagree	2	***Very Seldom***	2
Moderately Disagree	3	***Seldom***	3
Neutral	4	***Neither Seldom Nor Often***	4
Moderately Agree	5	***Somewhat Often***	5
Agree	6	***Often***	6
Strongly Agree	7	***Very Often***	7
Excellent	5	*None*	0
Good	4	*Very Few*	1
Fair	3	*Few*	2
Poor	2	*A Fair Amount*	3
Very Poor	1	*Quite a Bit*	4
Not Applicable	—	*A Lot*	5
Not Convenient at All	1	***Very Difficult***	1
Highly Inconvenient	2	***Difficult***	2
Inconvenient	3	***Not Difficult***	3
Neutral	4	***Easy***	4
Somewhat Convenient	5	***Very Easy***	5
Convenient	6	*Yes*	1
Highly Convenient	7	*No*	0

55.2.3.2.1 Survey Validity and Validity Determination

While the reliability of a survey is determined after the survey data have been collected, the validity of the survey has to be determined prior to the implementation. As the name implies, validity of a survey is the degree with which the survey instrument is valid in the results it produces or, in other words, whether the survey is measuring what it says it is measuring (Litwin 1995). Generally, within the context of HCI, validity is covered twofold: *construct validity* indicates the degree of how much the survey is backed by previous research in its field, how solid its construct is. In general, as in every research, development of a survey needs to rely on previous research to give the tool literature backing, proving that the survey didn’t come out of the imagination of the designer, but rather relies on a number of different research studies conducted by a number of different researchers. To prove the construct validity of their survey, designers need to prove the case that the questions they put into their survey are based on previous literature. Hence, in survey design, it is imperative to ensure that a majority of the questions have been implicated in the previous literature as items relevant to the current topic of interest. Without being able to prove this validity, it is not possible to make a convincing case regarding whether the survey is doing an undisputed contribution to the overall research topic of interest. However, it should be noted that it is almost impossible to provide a survey in which every single item has a full set of articles or books backing it. In most cases, some survey questions may have some indirect mention in the previous literature, and some survey questions may be solely based on the individual experience and/or opinion of the survey designers. This type of question generation is also acceptable, as long as the designers can prove that those questions are also based on solid research. In short, construct validity of a research survey in HCI aims at proving the conclusion unarguably that results obtained from this survey are on target and valid. Validity is therefore crucial to the success of the research conducted, of which the survey is a part.

Predictive validity is, simply put, the ability and power of the survey to predict correct results in repetitive use. A survey with predictive validity indicates that the results obtained from it in the current and future uses have the power of predicting accurate results. For example, if a comprehensive survey has been produced measuring the usability of a website, researchers will need to prove as part of their study that once this developed instrument is administered, the results that are produced accurately reflect the usability level and attributes of a website. Additionally, the survey also needs to accurately reflect usability levels and attributes when it is administered on other participant groups for evaluation of other sites. If these capabilities of the survey can be proven by the researchers, then the survey can be said to have predictive validity.

Like construct validity, predictive validity does not have any quantitative measurement method to be used. Hence, a survey's predictive validity again relies on qualitatively proving that the survey results are based on solid research notions, and hence the results are accurate. Survey designers need to explain that the results produced from their surveys have been proven to accurately reflect current situations concerning the target population's specifications, evaluations, and opinions, and will continue to do so in future applications when it is administered again. For this purpose, the elements of the survey need to be proven as elements that accurately predict results concerning the topic in focus. To make accurate predictions, surveys need to consist of elements that make accurate predictions themselves when their results are analyzed, and to ensure that these elements have predictive power, they need to rely on accurate literature findings and real-life situations. In short, similar to construct validity, predictive validity of a survey can be accomplished by ensuring that the survey relies on solid previous literature and the researchers' findings. Sometimes, to ensure that survey findings and questions have predictive power, they are evaluated by experts in the area prior to the implementation. Pilot testing is addressed later in this chapter.

55.2.3.2.2 Survey Reliability and Reliability Measurement

Reliability of a survey is the measure of whether the survey is measuring things consistently, and whether the results obtained from the survey can be relied upon. A survey's reliability affects its validity, as a survey that is not reliable in its measurements cannot produce fully valid results.

While there are a number of quantitative reliability measurement techniques for survey design, especially in psychology and sociology, the two most common reliability measurement techniques used in HCI research are *internal and inter-rater* reliability techniques.

The internal reliability technique is concerned with whether the survey questions are understood by the participants the way they are intended to be understood when they were prepared by the survey designers. An internally reliable survey contains questions that are all comprehended the same way by all participants at all times in repeated measures when it is administered. A lack of internal reliability is a common phenomenon, as different participants can understand survey questions differently if they are not asked in a highly clear and unambiguous fashion. Therefore, to improve the internal reliability of surveys, designers need to make sure to use statements that are entirely clear and leave no room for interpretation on what is meant in the questions. An example of a low internal reliability survey question would be "Did you have tremendous difficulty completing the tasks on the web page?" In this question, participants who had little difficulty, no difficulty, and a moderate amount of difficulty may respond to the question in a very similar way, resulting in confusion regarding whether the tasks were difficult or not. Additionally, if participants had difficulties in some tasks and no difficulties in the others, a question such as this may confuse the participants on what types of tasks (difficult or not difficult) they should base their response on. Obviously, in survey design it is important to be careful not to confuse the participants while they are filling out the surveys. Potential confusions can mostly occur on the participants' parts regarding what is meant by the survey question, and what the survey question is about (Cronbach 1990). Surveys may have high construct validity, meaning they may have been designed based on solid research, but if they confuse the participants with their questions, they will obviously lack internal reliability and, consequently, predictive power.

The most commonly used measure for internal reliability of surveys is called "Cronbach's Alpha Internal Reliability Coefficient" (Cronbach 1990). The coefficient relies on checking whether participants respond to the same question the same way when it is asked the second time, in a similar form. These types of questions are called "duplicate questions." The Cronbach's Alpha Coefficient is a correlation coefficient that determines the correlation between the duplicate questions, thereby giving an indication of whether the participants have the same understanding of a question when it is asked in a slightly different way, more than once. In many cases, the opposite, inverted form of the same question can be asked later in the survey. An example of two duplicate questions would be one question early on in the survey such as "Did you find the web design effective to complete the tasks?" and later, toward the end of the survey, "Was the web design effective to complete the tasks?" Alternatively, a question asking the same issue of web effectiveness in a reversed manner can also be posed later in the survey in an inverted question such as "Did you find the web design ineffective to complete the tasks?"

In general, one or two duplicate question pairs are put into surveys of moderate size, up to 40 questions. It may be more helpful to insert more than one pair of duplicate questions into surveys that contain more than 40 questions. Also, if the survey has sections (for example, in a survey measuring web usability, sections may include usability of colors, layout, navigation, etc.) it is recommended to have one duplicate pair of questions for each of the sections to have freedom about determining the individual internal reliabilities of each section.

The Cronbach's Alpha Coefficient is a correlation coefficient that produces a value between zero and one. The correlation between the duplicate questions is measured, and if the coefficient is equal to or greater than 0.7, then a survey is accepted as having high internal reliability (Cronbach 1990). A set of duplicate questions, another set of duplicate, inverted questions, and a sample Cronbach's Alpha Coefficient computer output from the SAS (Statistical Analysis Software) computer package are presented in Figure 55.4.

Internal reliability cannot be measured for surveys that contain open-ended questions. In empirical HCI research, however, most surveys with quantitative parts are required to have a satisfactory internal reliability coefficient in order to prove the reliability and validity of their results. Simply

Duplicate Pair with Same Question:

1. How would you rate the convenience of this interface?

○	○	○	○	○
Very Inconvenient	Inconvenient	Neutral	Convenient	Very Convenient

2. In general, how convenient was this interface for you to use?

○	○	○	○	○
Very Inconvenient	Inconvenient	Neutral	Convenient	Very Convenient

Duplicate Pair with Inverted Question (Inversion of the Scale Needed for Second Question):

1. How easy were the tasks?

○	○	○	○	○
Very Difficult	Difficult	Not Difficult	Easy	Very Easy

2. How difficult were the tasks?

○	○	○	○	○
Very Easy	Easy	Not Difficult	Difficult	Very Difficult

Sample SAS Output:

```
   Variables      Alpha
   Raw            0.881884
   Standardized   0.882042

Cronbach Coefficient Alpha with Deleted Variable

                                Raw Variables          Standardized Variables
Deleted                          Correlation                Correlation
Variable                      with Total     Alpha      with Total     Alpha
Question 1                    0.717499         .        0.717499         .
Question 2                    0.717499         .        0.717499         .

          Pearson Correlation Coefficients, N = 272

          Prob > |r| under H0: Rho = 0
               Question 1   Question 2
          Question 1   1.00000   0.71750
                       <.0001
          Question 2   0.71750   1.00000
                       <.0001
```

FIGURE 55.4 Sample duplicate questions and cronbach's alpha internal reliability coefficient SAS computer package output.

put, lack of internal reliability may result in the questions not measuring what they are intending to measure. Therefore, it is imperative in HCI research to insert at least one pair of duplicate questions into quantitative surveys.

More controversial than the internal reliability measure, the inter-rater reliability of a quantitative survey is concerned about the consistency among responses given by different participants to the same question. One argument is that in objective surveys, a consistency should be expected to some level among participant responses given to the same question in the same survey. While this argument may stay true to some extent in evaluation surveys, opinion surveys, as the name indicates, are about participants' opinions, which will obviously differ from person to person. Hence, it is arguable that the inter-rater reliability coefficient is a valid measure in opinion surveys. Additionally, a certain amount of variability is always present among responses to evaluation survey questions, even if the participants are all exposed to the exact same environment prior to the implementation of the surveys. The inter-rater reliability coefficient is a correlation coefficient among these responses given to the same question by different participants. And expecting a correlation as high as 0.7 among the participant responses may in most cases not be very realistic as a proof of reliability of a survey. Hence, while the inter-rater reliability coefficient is used in a number of survey types primarily in psychology, it is not seen as an essential measurement coefficient for survey reliability in relation to HCI research (Ozok and Salvendy 2000). The inter-rater reliability coefficient is also used to determine how professionals in psychology and sociology rate the same situation, but this type is not covered in detail due to its lack of relevance to HCI.

Measuring the designed survey's reliability is crucial to producing useful results in HCI research. Therefore, comprehensive survey designers must pay attention to these reliability measures while designing their surveys, ensuring that the results obtained from their surveys in current and future studies will have high reliability, thereby improving the theoretical and practical impact of their research.

55.2.3.2.3 Other Survey Evaluation Issues

A large part of evaluation of the surveys usually happens after they are administered. However, in most cases an equally crucial evaluation of a survey happens just prior to the implementation. This evaluation is the actual *pilot testing* of the survey, sometimes also referred to as "pre-testing" (Dillman 2000). As is the convention with most experimentation techniques in HCI research, the near-complete surveys can be administered to a group of participants. After this initial administration, the participants can be asked some questions about positive, negative, and missing issues in the survey, and any questions or sections that were incomprehensible or unclear. Based on the feedback, the administrators can revise the survey and prepare it for the final administration. While there are no firm guidelines regarding the number of participants the pilot testing of surveys should be run on, in most cases a minimum of three participants is recommended for moderate-size surveys (less than 200 participants). For large-size surveys, up to 10 participants are generally useful (Dillman 2000), although there is no upper limit for the number of participants to be used for the pilot study. Most surveys require revision after the pilot study, as in most cases there are some points the survey designers miss without the perspective of actual survey participants. In rare cases when no revisions are made to the surveys after the pilot survey administration, data obtained from the participants in the pilot can be included in the actual participant data pool.

How well a survey is designed is directly related to the validity and reliability of the results the research produces. Hence, the evaluation techniques covered in this section are crucial to the overall success of the designed survey and the research itself.

55.2.3.3 Survey Design, Redesign, and Revision

Design, redesign, and revision procedures for surveys to some extent bear some similarities to product design, redesign, and revision procedures. The initial design of surveys, as explained earlier, consists of generating questions based on the literature, expert opinion, and heuristics. Redesign and revision procedures mostly rely on the implementation of the survey on the entire group or a subgroup of participants. In most cases after a pilot test, surveys need revision which consists of changing or revising questions, scales, or instructions in the survey to make them clearer and more understandable for the participants. If there are errors in question structures or spelling errors, those are also located and eliminated after the pilot study. In rare cases, the required changes may be significant to the level that the survey may need redesign through the revision and change of most questions, scales, and instructions. It can be said that most small-scale survey revisions happen based on the feedback obtained from the pilot study.

Showing a certain amount of similarity to consumer products, frequently used surveys also need redesign and revisions over longer periods. Specifically in HCI, user habits and evaluation criteria for technology and technology-related products and issues change. It is therefore recommended that validated surveys that are used as empirical measurement tools in HCI should be reevaluated and updated about once a year to ensure they are up-to-date measurement tools and contain the latest additions to the HCI area as far as evaluation and opinion elements and techniques are concerned.

55.2.3.4 Illustrative Survey Design Example

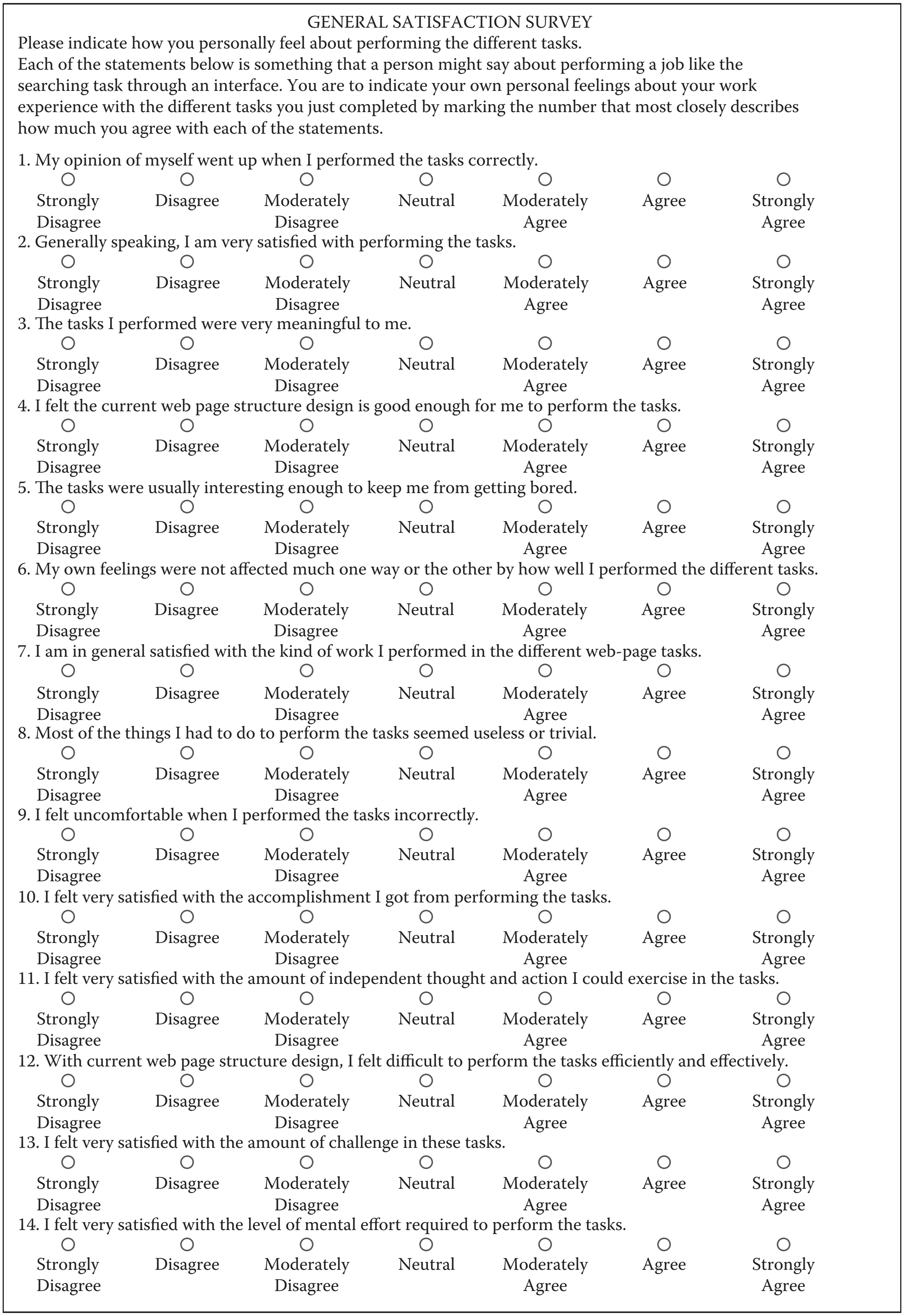
Figure 55.5 presents a sample of a complete, generic paper-and-pencil survey in the example of a postexperimental task satisfaction survey. In the design of a survey measuring the Tablet PC usability issues among academic professionals, the first step is to develop a literature portfolio. This portfolio should cover literature on both mobile computer usability and Tablet PC usability. Next, researchers may send an inquiry to a group of academicians who use Tablet PCs, inquiring about major usability categories in relation to Tablet PCs in open-ended questions. Based on the input from literature and expert opinions, the researchers create an initial set of questions and scales, pilot-test it, and administer the survey, most likely in an environment where they give specific Tablet PC tasks to participants in a controlled environment, preceded by the actual survey administration.

55.2.3.5 Challenges in Survey Design

Survey design challenges mostly deal with possible mistakes in producing the survey questions and scales. Additionally, some problems may occur due to the questions having no validity backing. Therefore, the key for HCI survey researchers is to gain strong background in the area through literature and expert opinion before designing the surveys. After this background is gained, researchers are likely to have no difficulty designing the surveys with the appropriate number and content of questions and scales, and a comprehensive set of instructions to be presented to the participants on how they should fill out the survey.

55.3 SURVEY IMPLEMENTATION IN HCI

Survey implementation can be categorized into open and controlled survey implementation environments. In this section, these two environments are first discussed, followed by



GENERAL SATISFACTION SURVEY

Please indicate how you personally feel about performing the different tasks.

Each of the statements below is something that a person might say about performing a job like the searching task through an interface. You are to indicate your own personal feelings about your work experience with the different tasks you just completed by marking the number that most closely describes how much you agree with each of the statements.

1. My opinion of myself went up when I performed the tasks correctly.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

2. Generally speaking, I am very satisfied with performing the tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

3. The tasks I performed were very meaningful to me.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

4. I felt the current web page structure design is good enough for me to perform the tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

5. The tasks were usually interesting enough to keep me from getting bored.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

6. My own feelings were not affected much one way or the other by how well I performed the different tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

7. I am in general satisfied with the kind of work I performed in the different web-page tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

8. Most of the things I had to do to perform the tasks seemed useless or trivial.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

9. I felt uncomfortable when I performed the tasks incorrectly.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

10. I felt very satisfied with the accomplishment I got from performing the tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

11. I felt very satisfied with the amount of independent thought and action I could exercise in the tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

12. With current web page structure design, I felt difficult to perform the tasks efficiently and effectively.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

13. I felt very satisfied with the amount of challenge in these tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

14. I felt very satisfied with the level of mental effort required to perform the tasks.

○ Strongly Disagree ○ Disagree ○ Moderately Disagree ○ Neutral ○ Moderately Agree ○ Agree ○ Strongly Agree

FIGURE 55.5 Sample of a paper-and-pencil satisfaction survey.

a discussion of sample representativeness issues, an implementation example, and discussions of implementation challenges and emerging and future trends in survey design and implementation.

55.3.1 Open versus Controlled Implementation Environments

Survey implementation (also referred to as "survey administration") occurs in two alternative environments. In controlled survey implementation environments, participants fill out the survey in an environment specifically designated for and arranged according to their activity. An open implementation environment, on the other hand, does not contain any specific environmental traits for the participant to implement the survey. Open environments also mostly do not include any restrictions on time or other factors.

Controlled environments for implementation of surveys usually have the goal of preventing any distraction for the participant to hinder his or her understanding or judgment. Controlled survey environments are in most cases well-lit experimental rooms with appropriate equipment to make the participant moderately comfortable (often consisting of a chair and a table). Both computer and paper-and-pencil-based surveys can be implemented in either open or controlled environments. If the survey is implemented in a controlled environment in front of a computer, the survey implementers need to make sure that the computer's alignments (screen brightness, glare, screen distance, keyboard height, and other ergonomics issues) are optimized for the participant. In controlled environments, in most cases a survey administrator is also available to answer possible questions from the participant. These types of controlled survey administration environments are usually used to implement evaluation surveys as in most cases participants had just completed computer-based tasks and for them to be able to evaluate the interfaces or any other HCI-related environments, products, or procedures, controlled environments force them to do those evaluations immediately, while the memories of the items for them to evaluate are still fresh in their minds. Opinion surveys are generally not implemented in controlled environments. In some rare cases in which recording participant behavior during survey implementation is part of the experimentation, a controlled environment can provide the equipment to nonintrusively record participant behavior/activities during the implementation of the survey.

While surveys are in some cases implemented in closed environments, doing so may be costly and time-consuming. Therefore, unless there is explicit need for a controlled environment, surveys are more commonly implemented in open environments. Open environments are environments of the participants' choosing, in most cases environments from their daily lives. In open environments, a survey administrator is not present. Implementing surveys in open environments has the advantage of giving the participants the choice to choose the time and place of the implementation. This flexibility above all increases the ease of finding participants. Additionally, the freedom for the participants to fill out the survey at their convenience also improves their feeling of freedom and may increase their enthusiasm, thereby improving the accuracy of the survey results (Dillman 2000). On the other hand, the surveys being filled out without the presence of a survey administrator will prevent the participants from asking any questions regarding the survey during the administration. Additionally, previous research has indicated that if participants have no particular motivation to fill out the survey, they may complete it very quickly without paying much attention, resulting in inaccurate survey results (Cochran 1977).

Both controlled and open survey implementation environments have advantages and disadvantages. In most cases, however, open environments are faster and more convenient to collect the needed data due to the flexibility they offer to both the participants and implementers.

55.3.2 Sample Representativeness in Survey Implementation

To ensure the validity of the results obtained from surveys, it is imperative to choose a representative sample of the target population to successfully implement the survey. It is common knowledge that the validity of the survey results improves with larger sample sizes. Therefore, researchers need to carefully choose both the sample sizes and the sample participants.

There are no strict rules for determining sample sizes in survey implementation. The size of the survey sample, meaning how many participants should fill out the survey, depends on the type of the survey and survey questions, as well as the number of questions in the survey. Thiemann and Kraemer (1987) summarized the statistical methods of determining sample sizes based on the number of variables being measured in the experiment. In most cases, surveys measuring general topics (for example, surveys about cell-phone-use habits of a certain population, such as college students) should be implemented on relatively large sample sizes, possibly no less than 60 participants for a survey consisting of up to 30 questions. For survey implementation, as a rule of thumb, the number of participants should always be bigger than the number of questions in the survey. While there are no set rules on choosing sample sizes for survey implementation, large sample sizes always improve the probability of obtaining high validity of surveys. For more on how to calculate optimal sample sizes in survey implementation to obtain satisfactory statistical power, see Thiemann and Kraemer (1987).

When surveys are administered, one of the most critical issues is to administer the survey on a balanced, nonbiased, and representative sample. In general, for surveys administered online, the survey should be sent to as big a potential participant pool as possible to ensure heterogeneity. A large group of potential participants should also be sought if the surveys are paper-and-pencil-based, by, for example, mailing out a large number of paper surveys. A balanced sample size in terms of race and gender ensures higher validity as well as a broader application of the results, and this can be achieved by sending the survey to a large base of participants. The demographic information collected also helps determine exactly whether the participant sample accurately represents the target population. As in any experimental or survey-based research, heterogeneity of the sample size allows the researchers to strengthen their argument that the results of their study are applicable and generalizable to the majority of their target population. For example, if there is a vast majority of males compared to females in the sample while the gender distribution in the actual target population is estimated to be about even, then the validity of the results may be argued upon as the variation of the results due to the females would not be taken into consideration in the population, and consequently in the conclusions derived from the survey results. Therefore, survey participant pools should be chosen carefully, and they should also adequately represent the target population. Especially in large sample sizes, sample characteristics may vary greatly, especially concerning age, education, and occupation demographics. In these cases, the demographics should be presented in detail as part of the research results. In most cases, a detailed explanation of demographics concerning participants' education and occupation can justify the representativeness issue of survey results, as long as the levels of these attributes do not differ very greatly between the sample and estimated target populations. In those cases, the results of the survey findings should indicate that the findings are likely to apply to the particular segment of the population that had an overwhelming majority among participants in the sample size. For example, suppose a survey on habits of the general population concerning the use of cell phones has been conducted. If the vast majority of the participants (more than about two thirds) are university students, then the researchers should indicate in their report of results that they measured the cell phone habits of the university student population, which constitutes a large percentage of avid cell phone users (Ozok and Wei 2004).

Survey participant pools require caution when they are chosen, and in cases when the researchers are convinced that the sample is not highly representative of the target population, they need to make clear that the results of their survey may possibly have a narrower focus. In most cases, such narrowing of the target population that the research is aimed at does not result in validity problems, but rather makes clear which population or population segment the results of the survey study apply to.

55.3.3 Challenges in Survey Implementation

As indicated in previous sections, surveys are a relatively easy way of collecting data. They usually don't require expensive equipment for implementation, and with the advent of the Internet, can be easily distributed, filled out, returned, and analyzed. However, there are still some serious challenges in survey implementation in both paper-and-pencil and electronic environments.

Looking at the big picture, surveys are sometimes referred to as "subjective measurements." While the author of this chapter strongly disagrees with this statement, the distinction should be made between measurement environments where performance measurements are taken objectively and unequivocally through camera recordings and software, and environments where participants are asked to indicate what is going through their minds. In the latter environment, obviously, there is no way to ensure that participants are putting on the surveys exactly what they think about an issue, an environment, a tool, a product, or a procedure. It is not uncommon that participants fill out a survey without paying much attention, or even randomly mark responses without reading the questions. The survey-reliability measurement techniques to some extent prevent this type of random data from being used. For example, whether participants took the survey questions seriously can be determined by looking for discrepancies between Cronbach's Alpha duplicate questions. Additionally, strict instructions given to the participants at the beginning of the survey in written or spoken format can also to some extent improve the probability of participants taking the time to read the questions and give replies carefully. Researchers using surveys as their primary research tool always reserve the right to eliminate participant data that look ill fated or incomplete. However, the researchers need to have evidence in their hands that the participant did not complete the survey by obeying the rules that were presented to them, not on any other ground such as the participant responses not being in accordance with the majority of other participants or with the direction of results that are expected from the research.

Another challenge is the "return rate" of surveys. Response and return rates among surveys that do not offer any compensation are extremely low—less than 20% (Dillman 2000). HCI research may involve lengthy surveys as well (more than 30 questions) which can potentially result in even lower return rates. Therefore, it is recommended that some sort of compensation should be offered to participants in survey research, whatever the resources will allow. This compensation may be small gifts (e.g., a pen or notepad), gift certificates, or cash compensation. Any of these types of incentives will surely improve the return rates of surveys (Miller 1996). Additionally, surveys can contain statements to convince participants that they will also learn important issues concerning the research while filling out the surveys. It should be noted, however, that the practical issue of finding subjects should not bias the sample, and recruiting of participants should be arranged according to the data-collection needs of the research, not according to what kind of participant groups are the most practical to collect data from. Recruitment activities need to be targeted to ensure a representative sample.

In most implementation activities, participants should be given their privacy while filling out the survey, ensured that their data will be kept confidential, and be provided a comfortable environment. Otherwise, they may want to either quit or finish as soon as possible without any consideration of the accuracy of responses. It should also be noted that surveys are voluntary, and therefore survey implementers should indicate the voluntary nature of the surveys and not pressure the participants. Research indicates that when participants are pressured to give accurate responses or when a mutual trust between the administrator and the participant is not established regarding the sincerity of both sides, they mostly produce very unreliable survey results (Dillman 2000).

Other survey implementation challenges involve participants' interaction with the survey interface. In paper-and-pencil surveys, the fonts on the paper should have enough size and familiarity for all participants, a pencil with an eraser head will allow participants to correct their responses, and survey elements such as questions and scales should be adequately distinct from each other to prevent any mistakes. Survey designers should use a very simple language in the surveys and avoid any little-known words and sentence structures (Gendall 1998). Surveys consisting of multiple pages should be clearly numbered on each page. This kind of a convenient interaction environment will improve participant enthusiasm and increase response rates as well as accuracy of survey results. Additionally, while some studies expressed concern about differences in survey responses among computer and paper-and-pencil surveys (Sanchez 1992; Cole 2005) and issues concerning the format in which online surveys are presented (Couper, Traugott, and Lamias 2001; Couper et al. 2004; Kiernan et al. 2005), a recent study indicated that the accuracy of survey responses did not significantly differ between surveys administered online and those administered paper-and-pencil (Carini et al. 2003).

In electronic surveys, the computer interface should be very simple and participants with little experience with computers should not have any difficulty using the interfaces. In most cases, participants are required to mark their responses with a mouse and type their responses on clearly marked, designated text spaces that have adequate size for visibility purposes. Some special equipment may offer some additional conveniences, such as touch screens. Screen glare and font sizes should be given consideration too. It can be recommended that computer surveys should be implemented on screens no smaller than 12 inches of diagonal size, with a refresh rate of at least 60 MHz. For surveys longer than one screen, scrolling should be minimized. It is recommended that each survey screen should not require more than two screen-heights of scrolling, and should be connected with hyperlinks or screen buttons, meaning once a participant completed a screen, he or she should be required to move on to the next screen by clicking on a screen button or a link. Besides the scrolling issue, if a participant sees a lengthy survey all presented on one screen, he or she may get discouraged to fill out the survey due to its length.

An additional potentially problematic item is the number of questions to ask participants in a survey. In most cases, the attention span of participants is very short, and surveys that do not offer any compensation are recommended to be shorter than 30 questions. In most cases, participants are not interested in spending more than 15 minutes in filling out surveys for which they don't get any compensation. There is always a trade-off between the size of surveys, meaning the ability to collect all the necessary data, and the ability to recruit subjects. Long surveys are more difficult to recruit participants for. Researchers should think carefully about compensation methods (money, gifts, gift certificates) if they intend to implement large-scale surveys.

Finally, Internet surveys also carry the potential of technical difficulties due to the variety of computers the participants may be using. Schleyer and Forrest (2000) identified usability problems, programming errors, and incompatibilities/technical problems as main problems identified in web-administered surveys. Therefore, Internet-based, especially web-based surveys, should not require any scripts or plug-ins to run and if possible should consist of simple Hypertext Markup Language (HTML) code, as HTML is universally recognized by all browsers.

These are the major challenges the survey implementers currently have to deal with. However, with careful design and implementation, as well as strict instructions containing comprehensive information presented to the participants regarding the survey, the challenges can be easily overcome, resulting in valid and reliable survey results.

55.4 EMERGING AND FUTURE TRENDS IN SURVEY DESIGN AND IMPLEMENTATION IN HCI

Surveys have retained their structure, more or less, for many decades. It can be said, however, that electronic and especially Internet-based implementation has changed the convenience level of survey implementation and data analysis in a significantly positive way. It is difficult to predict whether any new groundbreaking techniques will cause further leaps in survey design, development, and analysis, but if significant new developments will happen in the near future, they are likely to happen in the implementation technology. Internet surveys are on the rise, with the percentage of Internet-based surveys being on the rise for the past five years. While Internet-based surveys comprised 15% of all surveys implemented in 1999, this number increased to 70% in 2004, according to Nua Internet Surveys (2005). With the improvement in voice recognition and voice synthesis technologies, future surveys may eliminate the need of a visual interface; however, human-factors issues in these types of interfaces specifically for surveys are still to be explored. It is apparent that the number of surveys implemented through the Internet and other networks will continue to climb in the years to come, due to cost savings and a number of other convenience issues. In the future, HCI research is also likely to continue to use surveys as a main data-collection tool. With HCI research becoming a more integral part of technology design and creation (for example, human-factors engineers and software engineers working collaboratively in the software-design process), user surveys may become more integrated, collecting data for current or future users regarding both HCI and other technology issues. Additionally, computer literacy is increasing at a fast pace (U.S. Census Bureau 2005), which allows HCI survey researchers more freedom in asking more sophisticated questions concerning interface evaluation, current user trends, and more. All that said, the contribution of surveys to HCI research is highly significant these days, and will likely continue to be so in the many years to come.

REFERENCES

Aiken, L. R., and A. Lewis. 1996. *Rating Scales and Checklists: Evaluating Behavior, Personality, and Attitude,* 1st ed. New York: John Wiley & Sons.

Badre, A. 2002. *Shaping Web Usability: Interaction Design in Context*. Boston, MA: Addison Wesley Professional.

Baecker, R., J. Grudin, W. Buxton, and S. Greenberg. 1995. A historical and intellectual perspective. In *Readings in human-computer interaction: Toward the year 2000*, 2nd ed., ed. R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg, 35–47. San Francisco, CA: Morgan Kaufmann Publishers, Inc.

Baker, R. 1998. The CASIC future. In *Computer Assisted Survey Information Collection*, 1st ed., ed. M. P. Couper, 583–604. New York: John Wiley & Sons.

Belson, W. 1981. *The Design and Understanding of Survey Questions*. Aldershot, England: Gower.

Card, S. 1996. Pioneers and settlers: Methods used in successful user interface design. In *Human-Computer Interface Design: Success Stories, Emerging Methods, and Real-world Context*, ed. M. Rudisill et al., 122–69. San Francisco, CA: Morgan Kaufmann Publishers.

Carini, R. M., J. H. Hayek, G. D. Kuh, J. M. Kennedy, and J. A. Ouimet. 2003. College student responses to web and paper surveys: Does mode matter? *Res High Educ* 44:1–19.

Cochran, W. 1977. Sampling Techniques. 3rd ed. New York: John Wiley & Sons.

Cole, S. T. 2005. Comparing mail and web-based survey distribution methods: Results of surveys to leisure travel retailers. *J Travel Res* 43(4):422–30.

Converse, J. M., and S. Presser. 1986. *Survey questions: Handcrafting the standardized questionnaire*. Newbury Park, CA: Sage Publications, Inc.

Couper, M. P., M. W. Tourangeau, F. Conrad, and S. Crawford (2004). What they see is what we get:Response options for web surveys. *Soc Sci Comput Rev* 22(1):111–27.

Couper, M. P., M. W. Traugott, and M. J. Lamias. 2001. Web survey design and administration. *Public Opin* 65:230–53.

Cronbach, L. J. 1990. *Essentials of Psychological Testing*. New York: Harper & Row Publishing.

Dillman, D. 1991. The design and administration of mail surveys. *Annu Rev Sociol* 17:225–49.

Dillman, D. 1999. Mail and other self-administered surveys in the 21st century. *The Gallup Research Journal, Winter/Spring 1999* 121–40.

Dillman, D. 2000. *Mail and Internet surveys: The Tailored Design Method*. New York: John Wiley and Sons, Inc.

Gendall, P. 1998. A framework for questionnaire design: Labaw revisited. *Mark Bull* 9:28–39.

Hackman, R. J., and G. R. Oldham. 1980. Development of job diagnostic survey. *J Appl Psychol* 60(2):159–70.

Jenkins, C. R., and D. A. Dillman. 1997. Towards a theory of self-administered questionnaire design. In *Survey Measurement and Process Quality*, ed. L. Lyberg, P. Biemer, M. Collins, E. deLeeuw, C. Dippo, N. Schwarz, and D. Trewin, 165–96. New York: John Wiley & Sons.

Kiernan, N. E., M. Oyler, M. A. Oyler, and C. Gilles. 2005. Is a web survey as effective as a mail survey? A field experiment among computer users. *Am Eval* 26(2):245–52.

Krosnick, J. A. 1999. Survey research. *Annu Rev Psychol* 50:537–67.

Litwin, M. 1995. *How to Measure Survey Reliability and Validity*. Thousand Oaks, CA: Sage Publications.

Lund, E., and I. T. Gram. 1998. Response rate according to title and length of questionnaire. *Scand J Public Health* 26(2):154–60.

Medsker, G. J., and M. A. Campion. 1997. Job and team design. In *Handbook of Human Factors and Ergonomics*, 3rd ed., ed. G. Salvendy, 450–89. New York: Wiley.

Miller, K. 1996. *The Influence of Difference Techniques on Response Rates and Nonresponse Error in Mail Surveys*. Unpublished Master's thesis, Western Washington University, Ballingham, WA.

Myers, B. 1996. *A Quick History of Human Computer Interaction*. Carnegie Mellon University School of Computer Science Technical Report. CMU- S-96-163 and Human Computer Interaction Institute Technical Report. CMU-HCII-96-103, August, 1996.

Myers, B., J. Hollan, and I. Cruz. 1996. Strategic directions in human computer interaction. *ACM Comput Surv* 28(4):794–809.

Nielsen, J. 1989. Coordinating user interface for consistency. *SIGCHI Bull* 20:63–5.

Nielsen, J. 1993. *Usability Engineering*. London, UK: Academic Press.

Nua Internet Surveys. 2005. nua.com (accessed September 15, 2005).

Ozok, A. A., and G. Salvendy. 2000. Measuring consistency of web page design and its effects on performance and satisfaction. *Ergonomics* 43(4):443–60.

Ozok, A. A., and G. Salvendy. 2001. How consistent is your web design? *Behav Inf Technol* 20(6):433–47.

Ozok, A. A., and J. Wei. 2004. User perspectives of mobile and electronic commerce with a usability emphasis. *Proceedings of the ISOneWorld 2004 Conference*, Las Vegas, NE, Article 71.

Sanchez, M. E. 1992. Effect of questionnaire design on the quality of survey data. *Public Opin* 56:206–17.

Schleyer, T. K. L., and J. Forrest. 2000. Methods for the design and administration of web-based surveys. *J Am Med Inf Assoc* 7(4):416–25.

Shneiderman, B. 1992. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. New York: Addison-Wesley.

Stone, D. H. 1993. Design a questionnaire. *Br Med* 307(6914):1264–6.

Thiemann, S., and H. C. Kraemer. 1987. *How Many Subjects?: Statistical Power Analysis in Research*. Newbury Park, CA: Sage Publishing.

U.S. Census Bureau. 2005. Computer and Internet use in the United States: 2003. *Annual Report Special Studies* 23–108.

Zhang, Y. 2000. Using the Internet for survey research: A case study. *J Am Soc Inf Sci* 51(1):57–68.

56 Inspection-Based Evaluations

Gilbert Cockton, Alan Woolrych, Kasper Hornbæk, and Erik Frøkjær

CONTENTS

Usability inspection methods (UIMs) are approaches to usability evaluation based on expert inspection of a user interface and the probable user interactions with it. They can be applied to any designed artifact during development: a paper prototype, a storyboard, a working prototype (e.g., in Macromedia Flash™ or in Microsoft PowerPoint™), tested production software, or an installed public release. They are *analytical* evaluation methods, which involve no typical end users, unlike empirical methods such as user testing. UIMs require only availability of a designed artifact, trained evaluators, and supplementary project/evaluator resources. The resource requirement for evaluation is thus low: UIMs were one of the first groups of *discount methods* within human–computer interaction (HCI). Their origins as discount methods are important. Their inventors focused on reducing the cost of usability evaluation to a level that they judged to be acceptable for software development projects in the early 1990s. This focus on cost was at the expense of critical reflection and systematic scientific evaluations. Perversely for a user-centered field such as HCI, there was limited consideration of how evaluators would actually use methods, both *cognitively* from a perspective of problem-solving behaviors, strategies and tactics, and *socially* from a perspective of usability evaluation as *work* within project *contexts*. More recently, HCI has focused more on the affective aspects of interaction, and these too now need to be considered when assessing UIMs. It is not enough for UIMs to be better assessed and supported by better advice on their use by evaluators within usability work contexts. UIMs must also "feel right" to evaluators. It is important that evaluators believe that UIMs are helping them to find important usability problems and to understand and apply them well enough to be able to recommend effective design changes to remove the problems. It is unrealistic to expect UIMs alone to guarantee high quality evaluation. Interaction design is a complex challenging activity that demands extensive expertise and understanding.

This third version of this chapter takes a more critical perspective than previous ones, drawing on recent research in the context of 4 years of collaboration between researchers from over 20 European countries (the MAUSE project on Maturity of Usability Evaluation, www.cost294.org). At the heart of the problem with unrealistic expectations for UIMs is the word "method" itself, which hides many pitfalls for the unwary. Clearly, no published generic method can be complete. For example, accounts of user testing "methods" may give guidance on recruiting and screening test users and on designing and facilitating test tasks, but these accounts will never tell you who to recruit or which criteria to apply when screening potential test users, nor will they tell you exactly which tasks to facilitate. No account of user testing could possibly do this, because different interactive systems are designed to support different tasks for different groups of users to different standards of work or other qualities. While this all should be obvious, it is worth reminding anyone new to a design or evaluation method that practitioners and researchers *must work to get methods to work*. Methods do not apply themselves, nor do they provide all the resources (e.g., project-specific participant screening criteria) required to apply them.

Methods only exist in the past tense: you know what your method *was*, but at the outset of usability work, you are highly unlikely to know what your method *will be*. At best you will have a firm outline, supported by extensive project-specific details, of how you will *approach* usability work. We prefer to use the word *approach* to "method." In much commercial work, clients ask about a potential supplier's overall approach, rather than for step-by-step details of the methods that they will use. We believe that "approach" is a more accurate term for project-independent work structures, that is, prefigured activity skeletons that must be configured for specific projects. As with all human activities, methods are achievements and can never be premonitions. Evaluators need to see through the commodification of resource bundles as "methods," and be realistic about their real value prior to configuration and augmentation for specific evaluation contexts.

This chapter introduces UIMs, presents several example methods, and highlights the limited resources provided by each. Risks clearly arise from these limited resources, but well-informed practices disproportionately improve evaluation performance, improving cost-benefit ratios. Well-informed usability practices rely on additional resources provided by specific project environments, including evaluators themselves. Thus, evaluators must complement UIM resources with their own. As a result, it is very hard to rank UIMs or to make firm recommendations for the superiority of one UIM over another, even in fairly specific circumstances. The reason for this is that actual performance with a UIM depends on how it is configured and used in practice. This chapter, therefore, focuses on what is required to turn the rough approaches of UIMs into effective methods.

56.1 DEFINITIONS AND CONTRASTS

A UIM is an analytic approach to usability evaluation based on expert inspection of a user interface and the probable user interactions with it. A UIM *provides* resources that can be *applied directly* to an interaction design artifact, and/or probable user interactions with that artifact, and does not require end user resources. A resource is any element of an evaluation method, for example, knowledge about interaction design, evaluation procedures, problem report formats, checklists, or problem severity criteria. *Direct application* contrasts UIMs with model-based methods, which are *indirectly* applied via design representations (or models), requiring construction of models and secondary application of analyses to designed artifacts. Using the GOMS (Goals, Operators, Methods and Selection rules) method, for example, a task model would be analyzed and the results would have to be reframed to address the actual design artifact. In contrast, UIMs such as heuristic evaluation (HE) directly identify design features that may cause user difficulties. UIMs require fewer resources than model-based methods and thus commit evaluators to less work. While the reduction in costs could be accompanied by a reduction in benefits, the increase in costs for model-based methods may not be justified by a commensurate increase in benefits. In short, there is no fixed connection between the resources provided by evaluation approaches and their cost-benefit ratios. Evaluators thus need to pay careful attention to planning, supported by continuous critical reflection on the effectiveness of specific usability work practices. We next briefly review three UIMs to informally identify resources associated with, and missing from, each.

56.2 THREE EXAMPLE USABILITY INSPECTION METHODS

56.2.1 Heuristic Evaluation

With HE (Nielsen and Molich 1990; Nielsen 1992, 1994b), evaluators inspect a system with a view to discovering breaches of heuristics, which are "rules of thumb," rather than focused usability guidelines or exact style conformance rules. Breaches of heuristics indicate possible usability problems. One example of a heuristic breach is when a file is downloaded from a website, or intranet, without visible feedback to inform them of download progress. The failure to provide feedback breaches Nielsen's *visibility of system status* heuristic.

Heuristics are HE's main resource. The 10 current heuristics (Figure 56.1) were derived from analysis of 249 known usability problems, which had been established from evaluation of

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

FIGURE 56.1 HE's heuristics. (From Nielsen, J. 1994. *Usability Inspection Methods*. John Wiley & Sons. With permission.)

11 different interactive systems (Nielsen 1994b). The objective was to produce a compact heuristics set that best covered a set of known usability problems. HE also prescribes a specific procedure for performing an evaluation. Nielsen (1994b) recommends that analysts go through the interface at least twice. The first "pass" allows the evaluator to get a "feel" for the system, that is, both the general scope of the system and the flow of interaction. In the second and subsequent passes, the evaluator can focus on specific elements of the user interface. This procedure is HE's second resource, but advice is not given on how to structure the first pass or on how to focus in the second pass and associate possible usability problems with breached heuristics. These two critical resources must be provided by evaluators and/or their project/organizational context.

56.2.2 Cognitive Walkthrough

Cognitive walkthrough (CW) (Lewis and Wharton 1997) is a UIM primarily concerned with learnability, initially developed to assess "walk up and use systems" and later extended to task-based assessment of more complex systems. In CW, evaluators first of all assess if the user can form appropriate goals (Question 1) and then assess if the user can choose the appropriate actions to achieve their goal (Questions 2–4). The four CW questions are the following:

1. Will the users try to achieve the right effect? Does the user know what to do? Is it the correct action?
2. Will the user notice that the correct action is available? Is the action visible? Will users recognize it?
3. Will the user associate the correct action with the effect to be achieved? The action may be visible, but will the user understand it?
4. If the correct action is performed, will the user see that progress is being made toward solution of the task? Is there system feedback to inform the user of progress? Will they see it? Will they understand it?

The four questions, and an associated outline walkthrough procedure, are CW's main resources. Assuming the appropriate goal has been formed, the evaluator then breaks the "goal" down into the component task steps required to successfully achieve it. At each step, the evaluator asks questions 2–4. A negative response to any of the CW questions indicates a *possible* usability problem. To identify *probable* problems, evaluators must form success or failure judgments based on the cumulative impact of possible problems for a task. CW provides no guidance on the selection of tasks or on procedures for deciding success or failure cases. As with HE, these two critical resources must be provided by other means.

56.2.3 Metaphors of Thinking

Metaphors of thinking (MOT) (Frøkjær and Hornbæk 2002, 2008; Hornbæk and Frøkjær 2004) is a psychological-based inspection technique developed explicitly to address users' thinking, in the belief that the importance of such thought affects interaction. The metaphors are intended to support both evaluators and designers in understanding the importance of human thinking in interaction and to "stimulate critical thinking." Figure 56.2 summarizes the five metaphors. The basic method of performing an MOT inspection is that evaluators define representative tasks and "walk through" the interface. Evaluators make note of any usability problems identified, typically through violating the criteria in each metaphor. The main resources provided by MOT are thus these five metaphors and the prescription for their use. As with HE and CW, all other resources must be provided from other project/evaluator sources.

Metaphor M1: Habit Formation is Like a Landscape Eroded by Water

This metaphor is based concerned with the human trait of habit forming. The example used to illustrate this metaphor is that of adaptive menus that prevent habit forming. Since the menu's position can change from when an item was previously selected, habit forming is almost impossible.

Metaphor M2: Thinking as a Stream of Thought

The benefit of this metaphor is explained by the fragility of "stream of thought" and how interruptions can severely impact on task completion time. The example to illustrate this is that concentration can be affected even by such useful tools as e-mail alerts and automatic spelling checkers that, despite their respective benefits, can be distracting.

Metaphor M3: Awareness as a Jumping Octopus

This metaphor draws attention supporting users' associations with effective means of focusing within a stable context and, like CW, prompts consideration of whether users associate interface elements with the actions and objects that they represent. Users should be able to switch flexibly between different parts of the interface.

Metaphor M4: Utterances as Splashes over Water

This metaphor addresses support for changing and incomplete utterances, prompting consideration of alternative ways of expressing the same information, the clarity of interpretations of users' input in the system, and the risks of systems making a wider interpretation of users' input than users intend or are aware of.

Metaphor M5: Knowing as a Building Site in Progress

This metaphor is based on the notion that human knowing is incomplete, so users should not be forced by the application to depend on complete or accurate knowledge or to pay special attention to technical or configuration details before beginning to work.

FIGURE 56.2 Metaphors of thinking. (Adapted from Frøkjær, E., and K. Hornbæk. 2008. *ACM Trans Comput Hum Interact* 14(4):1–33.)

56.3 USABILITY WORK AND PRACTICE

Until recently, evaluation methods have been effectively viewed as if they were very similar to task methods in model-based approaches such as GOMS (Chapter 57), that is, evaluators follow a fixed predictable sequence of task steps to perform an evaluation. This has never been stated explicitly within evaluation method research, but many method comparisons assume that there will be a strong *method effect*, even when the *evaluator effect* (Hertzum and Jacobsen 2001) has been so clearly demonstrated and documented. In other words, comparisons have continued to assume that methods have strong effects even when it has been shown that evaluators are responsible for large variations in outcomes. Method effects require any differences in performance between evaluators to be solely due to the UIM being used. Gray and Salzman (1998) in their thorough critique of 1990s' evaluation method comparisons noted that the causal factor was more likely to be *evaluators using method X* rather than just *method X* alone. For the latter to be so, UIMs would have to be well defined and complete as to eliminate evaluator effects. Given the limited evaluation resources provided by HE, CW, and MOT, this is unlikely to be so.

The persistence of method effect assumptions in comparison studies reflects the origins of usability engineering in the *first wave HCI* of the 1980s, which was strongly influenced by cognitive psychology. The GOMS method, a first-wave HCI approach, used a very crude approximate model of expert human planning, which ignored two fundamentals of planning within HCI (Young and Simon 1987). First, the activity of planning is intimately interleaved with the execution of plans, and second, simple, partial plans are more appropriate than complex, detailed ones. Thus by the late 1980s, cognitive scientists such as Young and Simon had shown that viewing methods as rigid pre-scripted plans was wrong. Second wave HCI, with its “turn to the social” in the 1990s, reinforced these positions with rich accounts of human work that showed that plans were *resources* and not *scripts* (e.g., Suchman 1987). However, only in the last few years has HCI research on evaluation method usage finally treated usability practices as human work, with all the variations and contingencies that this involves. We now briefly review this delayed “turn to the social” in evaluation method research and next clarify its implications for our understanding of usability evaluation.

56.3.1 Realities of Usability Work

Some HCI research has continued to compare evaluation methods based on an unrealistic position on the power of methods, but usability specialists cannot base their work on such illusions. Extensive project-specific requirements (e.g., choosing participants, reporting results) distance methods in use from published methods, even when these superficially appear to be detailed step-by-step practitioners' “cookbooks.” Usability work does not get by through “choosing a method”; however well such a choice can be grounded. Instead, usability work *combines* several methods (Rosenbaum 2008), which generally compensate for each other's weaknesses, reducing or even removing the relevance of rankings or assessments of isolated methods. Molich et al. (2004) showed how teams that tested the same website differed on many specific choices on how to conduct a usability test. For instance, with respect to test reporting and task selection, marked differences were found. The nine teams not surprisingly found different sets of problems. Yet, very few scientific studies look at the combination of methods (though see Uldall-Espersen, Frøkjær, and Hornbæk 2008); this supports our argument that very few comparative research studies' evaluation investigates methods as they are used in practice. Rosenbaum's (2008) account of the evolution of usability practice shows a clear move over the last decade to project-specific combinations of multiple methods. To use a culinary analogy, usability work prepares meals, not individual dishes.

Method research in HCI should not primarily focus on supporting the choice of *a* method, nor should it attempt to answer questions such as what, in some specified context, is the “best” method, where “best” may mean most productive, most thorough, most valid, easiest to use, or cheapest to use, casting usability work as a simple choice of methods. Once chosen, the expectation is that evaluators will faithfully adhere to a method. Real world choices can never be specified wholly in terms of options and selections. There is more to choice than the menu. There is also the diner and their needs. While criteria such as nut allergies have straightforward consequences for choices from menus, wanting something “interesting and different” does not. The goodness of a choice lies in the context of the choice. Thus, Furniss (2008) argues that an appropriate choice and use of a method is functionally coupled to the project context, including client biases, practitioner expertise, their relationship, the budget, the problem, the time, auditing potential (for safety critical systems development), and persuasiveness.

56.3.2 Method as Achievement

It is over two decades since Suchman (1987)'s breakthrough critique of intelligent photocopiers. Suchman's research readily exposed how the plan recognition built into an intelligent photocopier was no match for the variety of real user behaviors. By treating a plan as a script, the intelligent photocopier was too rigid and unimaginative in its interpretation of user behaviors. Suchman argued that plans should be regarded as *resources* that have some fit to potential real world situations, but to achieve fit such plans need to be modified and extended to cope with the situated realities of human behavior. If we accept that it is highly unlikely that scripted plans in intelligent appliances can ever support management of situated human behavior, designers of usability methods face even bigger challenges, since usability work is far more challenging and complex than, for example, photocopying. We need to remove such inconsistencies within HCI thinking. If we cannot script user interaction, then we cannot script usability work. This has been shown for the user of user testing methods by Nørgaard and Hornbæk (2006), who studied how think aloud tests were conducted in seven Danish companies; they found that immediate analysis of observations made in think-aloud sessions was only performed sporadically, if at all. Usability approaches must then be designed, understood, and assessed within a framework that recognizes the situated nature of all human activities. Within such a framework, evaluation methods can never be anything more than weak prescriptions.

UIMs thus cannot be provided in any complete form by usability researchers or leading practitioners. Instead, what is provided are *approaches* that commoditize a set of resources by packaging them together under some method "brand name" (e.g., HE, CW). To return to a culinary analogy, what are called UIMs have much more in common with a Chicken Fajita *kit* on a supermarket shelf than with a Chicken Fajita *meal* in a restaurant. Consumers of the former expect it to be incomplete, and they expect to have to source some key ingredients and then prepare them, combine them with the kit, and cook them. Purchasers of the latter just expect to eat Chicken Fajitas and would be very dissatisfied if they had to prepare or cook anything. Evaluators who expect methods to be like completed meals will be similarly dissatisfied. They should expect *approaches* that require them to augment and adapt. With this in mind, we now further review UIM "ingredients," both those provided by "branded methods" and those developed within HCI research that can readily be combined within a range of evaluation resources.

56.4 RESOURCES FOR USABILITY INSPECTION

Attending to the resources present in (or absent from) descriptions of UIMs offers several benefits to researchers and practitioners. First, the important choices in usability inspection are among resources, not among methods. Second, resources may be configured and combined as evaluators see fit, which is overlooked when the focus is on methods. Third, some approaches to inspection require resources that are implicit or unusable, which a focus on resources can better expose. Fourth, focusing on resources allows sharper and more controllable research studies with a more realistic chance of benefitting practitioners. The studies can focus on specific aspects of usability work (resources) rather than incomplete combinations of resources (approaches).

56.4.1 Resources within Three Example Usability Inspection Methods

To exemplify the notion of resource in UIMs, we return to the three example UIMs outlined earlier in this chapter. First, for HE, we have already identified heuristics as the key resource. The two original papers on HE (Molich and Nielsen 1990; Nielsen and Molich 1990) listed nine heuristics (e.g., "simple and natural dialogue" and "prevent errors"). Subsequently, the heuristics have been through several iterations. Nielsen (1994b) did a post hoc analysis of 249 usability problems to find seven heuristics that could have predicted many of those problems and added three others to produce a more comprehensive set.

As noted earlier, HE also provides a procedural resource, but descriptions of it vary in their recommendation of the use of *task resources*. Originally, Nielsen and Molich (1990) mention no need for formal or informal understanding of tasks for evaluators. Later, Nielsen (1993) mentions that in contexts where evaluators lack *domain knowledge* (which Nielsen had shown to be a critical evaluation resource), they may be supplied with "a typical usage scenario" (p. 159) even though "evaluators are not *using* the system as such (to perform a real task)" (p. 159). Despite many studies of HE, we are unaware of any that have tried to manipulate task specification and analysis resources, even though Sears (1997) showed that combining CW's strict procedure with HE's loose sweep improves over the performance of HE alone.

In themselves, heuristics are not very usable. Both studies of, and practical introductions to, HE provide further tutorial material. Nielsen and Molich (1990) suggested that the heuristics could be presented to evaluators in "a single lecture" (p. 250). Nielsen (1994b) walked through a set of heuristics, explaining and exemplifying them almost 50 pages. Because of this variety in introductory material to HE, some experiments have used lectures and tailored material to standardize evaluator education (e.g., Cockton and Woolrych 2001, which identifies when task resources are essential to identify heuristic breaches). The extent to which such additional *knowledge* resources influence performance with HE is presently unclear.

Given the limited and varied resources associated with HE, it is no surprise that HE's resources alone cannot explain documented evaluation outcomes. Jeffries et al. (1991) asked evaluators note how they identified problems when using HE. Analysis of their notes showed that roughly a quarter of problems were found via "side effects" (e.g., from prior experience) and about a third of the problems were found "from prior experience with the system." Cockton and Woolrych (2001) demonstrated that many outcomes of HE could not be attributed to its resources, since evaluators successfully predicted usability problems with which no heuristic could be credibly associated, and thus had to inappropriately associate a standard heuristic with many of the one third of usability problem predictions that were confirmed by carefully focused user testing.

Turning to CW, as noted, its key resource is its prescribed procedure, based on a set of tasks, which answers four questions for each step of a task. In contrast to HE, CW's procedure is clear and always task-centered, although the final step of forming success/failure cases is not well described. CW forces use of task resources by requiring analysts to answer four questions for each task step, and then construct a success and failure case for the task. However, there is no support from CW for forming success and failure cases, nor is there support for task selection and specification. Even so, CW's success and failure cases are rare examples of an *analysis resource* for *problem elimination*, letting analysts argue that some apparent design flaws would not automatically cause severe usability problems. In contrast, most UIMs provide *discovery resources* that guide evaluators to find *possible* problems. To avoid a high proportion of *false negatives*, UIMs need to also provide *analysis resources* that support evaluators in deciding whether a found problem is *probable*, and thus should be reported, or *improbable*, and thus should be discarded.

The cost-benefit balance arising from CW's task-based procedures may be unfavorable. Early studies such as John and Packer (1995) found that going through CW was "very tedious." Early versions of CW required analysts to explicitly consider users' goals (e.g., Lewis et al. 1990; Polson et al. 1992). For example, the version presented by Polson et al. (1992) required three pages of questions for each action; one page to address users' goals, one to cover choosing and executing the correct action, and a final page of questions covering the effect of taking the action on the user's goal structure. The overhead of CW and difficulty applying it led to the cognitive "Jogthrough" (Rowley and Rhoades 1992) and multimedia tool support for CW procedures (Rieman et al. 1991), which increased evaluation efficiency markedly, that is, they improved CW's cost-benefit ratio by adding a further *tool resource*.

A simpler version of CW was developed by Wharton et al. (1994), which de-emphasized "the explicit consideration of the user's goal structure," resulting in the current set of four questions, although a subsequent variation (Spencer 2000) reduced these down to two! Overall, later versions of CW reduced extensive preparation materials for analysts, making them less unwieldy as discovery resources, but perhaps losing some benefits that arise with interaction-centered methods, with some benefits preserved via the use of task descriptions, but others lost due to the progressive simplification of CW and the resulting loss of explicit resources for evaluation support.

Early versions of CW emphasized *theory as a resource*. Answers to questions should be supported by empirical data, experience, or scientific evidence, all knowledge resources external to CW, except for CE+, a theory of learning (Polson and Lewis 1990) that provided CW's foundations. For example, CE+ predicts that users new to a system will choose an action with a good match to their current goal. This reflects the relation between interaction-centeredness and grounding in theory, in that the system-centered methods such as HE are generally atheoretical. Later versions of CW backgrounded the CE+ theory of learning (Wharton et al. 1994; Spencer 2000), suggesting that there is a tendency to favor concrete (notations, questions) over abstract resources (theories, concepts) and practical ones (e.g., procedures) over propositional resources (e.g., knowledge, information).

As noted for the third example UIM above, MOT's key resource is the five metaphors and their associated key questions (Table 56.1). Metaphor choice was guided by considerations of how this resource would work for analysts. MOT's inventors argued that "use of metaphors as a communication device supports intuition and requires active interpretation; an effort orthogonal to developing inspection techniques that are more strictly formal and piecemeal analytical" (Frøkjær and Hornbæk 2008, p. 3). The extent to which the metaphors and key questions actually work in this way for analysts is, however, only investigated through summary measures of evaluation performance and through analysts' comments on MOT.

The procedure of MOT is similar to that of HE (Frøkjær and Hornbæk 2008), but contains more detailed instructions on how to conduct an evaluation (Hornbæk and Frøkjær (2002). The instructions mention that the analyst must use "typical tasks" to drive the evaluation, but does not provide any resources for identifying these. MOT thus provides similar task resources to CW, that is, stating that they must be used, but providing no support for selecting and specifying them.

In the above analysis, we have begun to distinguish different classes of resources. Two broad classes are *discovery*

TABLE 56.1
Summary of the MOT-Technique: The Five Metaphors, Their Implications for User Interfaces, and Examples of Questions to Be Raised

Metaphor of Human Thinking	Implications for User Interfaces	Key Questions/Examples
Habit formation is like a landscape eroded by water.	Support of existing habits and, when necessary, development of new ones.	Are existing habits supported? Can effective new habits be developed? Is the interface predictable?
Thinking as a stream of thought.	Users' thinking should be supported by recognizability, stability and continuity.	Does the system make visible and easily accessible the important task objects and actions? Does the user interface make the system transparent or is attention drawn to non-task related information? Does the system help users to resume interrupted tasks? Is the appearance and content of the system similar to the situation when it was last used?
Awareness as a jumping octopus.	Support users' associations with effective means of focusing within a stable context.	Do users associate interface elements with the actions and objects they represent? Can words in the interface be expected to create useful associations for the user? Does the graphical layout and organization help the user to group tasks?
Utterances as splashes over water.	Support changing and incomplete utterances.	Are alternative ways of expressing the same information available? Are system interpretations of user in-put made clear? Does the system make a wider interpretation of user input than the user intends or is aware of?
Knowing as a site of buildings.	Users should not have to rely on complete or accurate knowledge—design for incomplete-ness.	Can the system be used without knowing every detail of it? Do more complex tasks build on the knowledge users have acquired from simpler tasks? Is feedback given to ensure correct interpretations?

and *analysis* resources, which respectively support finding possible usability problems and deciding on their probability and severity. *Task specification resources* are a subclass of discovery resources when used to guide problem discovery. However, information on *task frequency and criticality* can be regarded as analysis resources when they are used to determine the severity of a probable problem. Here, they would also be instances of *severity rating* resources that *rank* probable problems by their *likely impacts* on users. Other examples of evaluation resource include *matching and merging* resources (to *consolidate* the predictions of one or more evaluators into a single set of probable problems) and *expressive resources for reporting* (to support merging/matching and/or communication of the final problem set). Further types of evaluation will be identified in the next two subsections.

56.4.2 Resources within Other Usability Inspection Methods

We now introduce four further UIMs. The first, *Heuristic Walkthrough* (HW: Sears 1997), is a hybrid approach that combines the resources of HE and CW. HW also requires a prioritized list of user tasks, which should include frequent or critical ones, and may also include tasks designed purely to ensure coverage of the system being evaluated. HW thus provides more guidance on task selection than HE, CW, or MOT. This *discovery* resource is complemented by a further procedural resource that provides additional support for discovering possible problems by combining and extending the basic discovery procedures of CW and HE. It has two phases: task-based and free-form. In the first phase, evaluators explore tasks using a set of thought provoking questions derived from CW. They are free to explore the tasks in any order, spending as long as they need, but they should be guided by task priorities. In the second phase, evaluators are free to explore the system. They use the set of thought provoking questions *again*, plus HE's heuristics. Sears (1997) compared HW against CW and HE. HW found more actual problems than CW and had fewer false positives than HE. Overall, Sears' merge of HE and CW improved evaluator preparation and discovery resources, with evidence of improved analysis resources (reduction in false positives), which may be due to the initial CW inhibiting discovery of unlikely problems. HW appears to encourage more self-reflection on the part of evaluators relative to HE, but this is not due to the "method as a whole," but to a single procedural resource (two phase problem discovery structure).

Turning to our next UIM, *Ergonomic Criteria* (EC: Bastien and Scapin 1995; Scapin and Bastien 1997) uses a set of 18 ergonomic criteria. They were formed by reviewing around 800 existing guidelines and experimental results (Scapin 1990). Not surprisingly, HE's heuristics and EC are often very similar, for example, Immediate feedback (HE: visibility of system feedback) and Quality of error messages (HE: help users recognize, diagnose, and recover from errors). Each has a definition, a rationale, and examples of guidelines plus comments that include help to disambiguate any related criteria. There are eight main (top level) criteria: guidance, user workload, user explicit control, adaptability, error management, consistency, significance of codes, and compatibility. Five of these are further subdivided, resulting in 13 second-level criteria: prompting, grouping and distinguishing items, immediate feedback, clarity, brevity, mental load, explicit actions, user control, flexibility, users' experience management, protection from errors, error messages, error correction. Further subdivision results in overall 18 criteria at three levels of abstraction and structure.

While some accounts of HE provide similar support (e.g., Nielsen 1993), EC's consistent structures are rare for UIMs. EC's structure is likely to assist evaluator preparation for using it. EC thus considers evaluators' learning needs and provides structure and detail to prepare them for using it. Important preparation resources including scoping, axiological and knowledge resources (Woolrych, Hornbæk, Frøkjær, and Cockton 2012), which respectively indicate the scope of a UIM, the values underlying its design, and the knowledge required to fully understand it. Note that these resources have an impact before inspections are attempted in better *preparing* evaluators. Preparation resources are distinct from discovery or analysis resources, although both of the latter clearly interact with the former. EC also requires evaluators to be provided with description of the *purpose* of the product, a preparation resource that is lacking in HE, CW, MOT, and HW. EC also provides procedural resources that support discovery and analysis: a first phase follows a task-based walkthrough, while a second phase focuses on problem diagnosis.

Our reviewed UIMs so far have distinct motivations. HE focused on discounting evaluation costs, CW on learning walk-up-and-use systems, MOT on empathy with user behaviors, EC on supporting evaluator learning, and HW on trading off one evaluation cost (false negatives) against another (first pass walkthrough costs). Our next reviewed UIM, *Cognitive Dimensions* (CD: Green and Petre 1996), sought to improve evaluators' conceptual vocabulary, moving them away from technical surface features (Green 1991). CD attempts to capture a set of orthogonal dimensions for restricted set of systems called "notations," which are used to design information structures, for example, programming languages and spreadsheets. These dimensions are not necessarily criteria for a design but a characterization of the design space.

Sample of dimensions are shown in Figure 56.3. The dimensions embody a theory about how people use notations

Hidden dependencies occur when one cannot see how a value is calculated, for example, a spreadsheet where a small change can have unexpected effects. All dependencies that are important to the user must be accessible.

Premature commitment occurs when a user is forced to make a decision before the necessary information is available.

Viscosity is resistance to change or a measure of how much work is needed to achieve a change.

Abstraction gradient: An abstraction is the grouping of objects into one object, which can make a system easier to understand.

FIGURE 56.3 Example cognitive dimensions.

to build information structures. A simplified theory of action would involve people translating goals into specifications, then into actions, and executing them. Rather, people work bottom up, top down—goals and subgoals may be attacked at any moment. Green (1991) claimed the preferred strategy is opportunistic where high- and low-level decisions are mingled; commitment can be weak or strong and development in one area may be postponed because of problems foreseen in other areas. The vocabulary of dimensions may act as a discovery resource, sensitizing analysts to generic interaction difficulties. However, once associated usability problems are discovered, there appears to be little further support for problem confirmation or elimination. Even so, as with CW's CE+, CD provides *theoretical knowledge resources* that provide valuable *preparation* resources for evaluation, without which CD would provide little guidance on applying the dimensions. A notation called "ERMIA" (Green 1991) was proposed as a method of exploring dimensions. Interestingly, this notation seems to have been developed (Green and Benyon 1995, 1996) into a stand-alone UIM without dimensions. ERMIA is our second example of a tool resource (Rieman et al's automated CW was the first). Thus, as well as practical knowledge resources (list of cognitive dimensions), CD provides a tool resource that supports evaluation practices. The theoretical focus on opportunistic planning draws on work by Green's colleague Young at MRC Cambridge (Young and Simon 1987), and thus strongly resonates with the position on usability work in this chapter that evaluators do not systematically follow methods top down.

Our final reviewed UIM, *Pluralistic Walkthrough* (PW: Bias 1994), originated at IBM. In PW, an administrator leads a team of three diverse participants (a representative user, a product developer, and a usability specialist) who pretend to be system users. Each is given a set of hard copies of screens and task descriptions. For each screen, the participant is asked to write down in as much detail the next action (or set of actions) that the user would take to achieve the task and any comments they have about the screen. Once each participant has written down the next action, the screen is discussed. When the representative users have finished talking (e.g., about problems), the usability specialists and product developers talk (e.g., to explain the rationale of certain features). No assessment of this UIM has been published; however, there are clear potential benefits from *preparation* resources (hard copies of screens, task descriptions), and the combination of representative users, product developers, and usability specialists provides further discovery and analysis resources in the form of *knowledge of users* (from representatives), of the *product* being evaluated (from developers, who also provide *technical* background knowledge), and of *tasks* and *interaction* (from usability specialists). PW is distinct in adopting a *social* strategy to provide evaluation resources. Usability specialists draw up task descriptions, developers provide copies of relevant screens for the task, representatives provide user knowledge, and the administrator provides procedural knowledge of PW. In some ways, PW anticipated the recent and long overdue "turn to the social" in HCI evaluation research and practice, motivated by the opportunity to draw on evaluation resources that already exist within the project context. Unlike Spencer's (2000) restrictions on CW, PW treats project social interactions positively, rather than a potential negative source of disagreements.

56.4.3 Generic Resources for Usability Inspection

In addition to resources that are closely associated with particular UIMs, many resources are discussed outside the context of particular UIMs or are common to several. Table 56.2 summarizes some of the most important generic resources for usability inspection. First, *task selection and specification* are often key to preparation and provide a resource for many inspection and model-based approaches. As with individual differences, the selection of particular tasks has a large effect on the interaction with computers and hence on evaluation outcomes (Lindgaard and Chattratichart 2007). Sears and Hess (1999) described how the amount of detail in task description affected which problems were found in CW. Analysts who used detailed task descriptions found significantly more

TABLE 56.2
Generic Resources for Usability Inspection

Resource	Purposes	Examples
Task selection	Selecting and specifying tasks for inspection or user testing	Sears and Hess (1999)
Task walkthrough	Supporting inspection methods with ways of going through tasks and interfaces	Sears (1997)
Heuristics	Resources for discovering and thinking about usability defects	Hvannberg, Law, and Lárusdóttir (2007); Mankoff et al. (2003); Somervell and McCrickard (2005)
Reporting formats	Helping communicating problems and solutions for subsequent analysis, evaluation auditing, iteration, and customer communication	Cockton, Woolrych, and Hindmarch (2004); Capra and Smith-Jackson (2005); Theofanos and Quesenbery (2005)
Problem analysis and classification	Ways of classifying problems so as to analyze, merge, or reject them	Andre et al. (2001); Cuomo and Bowen (1992)
Problem merging	Identifying similar problems, for instance, through tool support or using different definitions of similarity	Connell and Hammond (1999); Hornbæk and Frøkjær (2008a); Howarth, Andre, and Hartson (2007)

low-severity problems, but fewer problems relating to feedback from the system, compared to evaluators who used less detailed descriptions. Similarly, obvious differences have been shown to exist in user testing between user-generated tasks and set tasks, and also between open-ended and closed tasks (Spool and Shroeder 2001), but little research has been conducted on the impact of task types within usability inspection.

Similarly, CW and related UIMs are the only ones to provide resources for guiding evaluative task walkthroughs. There is also considerable scope for research here to compare the costs and benefits of different task walkthrough procedures. For example, when using CW, once one of the four questions has been answered in the negative for one step, it may be the case that all subsequent task steps should have negative answers for the same question. Alternatively, the answer to a question may have to be deferred for several steps, until it can be confidently answered. This is especially the case for the fourth question: *if the correct action is performed, will the user see that progress is being made toward solution of the task?* This may not become clear for several steps after the correction action is performed. There is thus scope for streamlining task walkthroughs by propagating/deferring answers to later steps. However, the effectiveness of such streamlining needs to be demonstrated through research. The only relevant research here currently relates to the presence or absence of task walkthroughs (e.g., HE vs. HW, Sears 1997).

While HE provides the best known set of heuristics for usability inspection, *alternative heuristics* are possible: Hvannberg, Law, and Lárusdóttir (2007) have compared Nielsen's heuristics to Gerhardt-Powals' cognitive engineering principles. Mankoff et al. (2003) developed and tested a specialized set of heuristics for ambient displays. Somervell and McCrickard (2005) have developed similar heuristics for large-screen information exhibits through the use of a structured process that relies upon critical parameters. Muller et al. (1995) rightly claimed that Nielsen's (1994b) heuristics ignore usage contexts. They added three extra heuristics to assess how well a design fits user needs and the work environment:

1. Respect the user and his or her skills
2. Pleasurable experience with the system
3. Support quality work

These studies have shown how specific heuristic resources impact evaluation performance. Heuristic sets vary considerably in their generality and derivation (e.g., empirical for HE and EC, systematic focus for Somervell and McCrickard, theory for CD and Gerhardt-Powals), as well as in their structure and presentation (e.g., HE vs. EC). There is again considerable scope for practically oriented research that can isolate the varying impact of overall philosophies, systematic foci, structure and presentation, and formal derivation of heuristic sets. Effective innovation in any of these areas would improve evaluation performance using heuristic sets.

Task selection and task walkthrough procedures, plus heuristic focus, derivation and presentation are relevant to only UIMs that include or require them as resources. However, some resources are rarely provided or required by UIMs, and yet are unavoidable. For example, the way that usability problems are reported plays an important role in usability inspection, and thus report formats are potentially a key evaluation resource.

Cockton et al. (2003) demonstrated that an *extended problem reporting format* can positively impact evaluator performance on false positives and appropriate HE use, with results here replicated in Cockton, Woolrych, and Hindmarch (2004). This extended format required evaluators to explain how they found a possible problem and how they decided whether to confirm it as a probable problem or discard it as an improbable one. This appeared to have improved evaluators' self-awareness and repertoires when *looking for* possible problems, making problem discovery less tacit. It also appears to have improved self-awareness and repertoires when *analyzing* possible problems, encouraging them to carefully consider whether a possible problem should be treated as probable or improbable. More explicit behaviors appear to have been prompted by the need to explain problem discovery and analysis, confirming the wisdom of an earlier conjecture that having justifications for why something is a problem is important (Jeffries 1994); the downstream utility of problem descriptions seem to increase with such justifications (Hornbæk and Frøkjær 2006).

Structure and its content are generally created simultaneously for problem report formats, but adding a single content item to an existing format can significantly improve evaluation quality. For example, evaluators in a think aloud test were instructed to consider *business goals* while evaluating and to report, together with typical parts of problem reports, the importance of a problem in relation to business goals (Hornbæk and Frøkjær 2008b). Compared to a control group, this format made evaluators report fewer problems, but the problems reported were seen as being of more utility in the development process. Although the benefits of this report item were demonstrated for user testing, it is reasonable to assume that similar benefits would arise if used alongside UIMs and might be particularly useful for evaluators with limited knowledge of the business goals for an application. Such a small change with a potential large impact further illustrates the value of focusing UIM research on the impact of specific resources, rather than on the overall impact of highly variable incomplete "methods."

A further specific report format element is the *type* of the usability problem, for example, its severity class (e.g., Jeffries et al. 1991). Types may be based on a range of constructs, for example, by relation to stages in Norman's (1986) theory of action. Using such a coding scheme, a study (Cuomo and Bowen 1992, 1994) found that an early version of CW tended to focus on problems related to the formation of actions and was poor at finding problems concerned with evaluating the display. A similar coding approach was used for an assessment of a later version of CW (Sears and Hess 1999), which demonstrated the impact of short and (very) detailed task descriptions on analyst performance. The impact of each resource was assessed by coding problems by the CW question relevant to the prediction. Short task descriptions resulted in relatively more predictions related to finding actions (CW Question 2). Detailed descriptions (to the level of naming and locating controls) resulted in relatively more feedback-related predictions (CW Question 4). Sears and Hess (1999) explain the differences through the impact on how evaluators discovered problems. Detailed descriptions led to finding fewer action specification problems than short descriptions did, but left evaluators with the energy to look for feedback problems.

In most usability work, individual problems are predicted or discovered that are instances of more general and/or frequent user difficulties. Such individual problem instances need to be matched to reduce a large set of individual predictions or user difficulties to a master set. *Problem matching and merging* is a major source of confounds when comparing usability methods (Cockton and Lavery 1999; Connell and Hammond 1999; Hornbæk and Frøkjær 2008a; Lavery, Cockton, and Atkinson 1997). Hornbæk and Frøkjær (2008a) showed how differences in instructions for how to match usability problems significantly affected the number of problems that could not be matched with other problems (in a set of 55 problems): The degree to which a single problem type emerges from UIM use matters a lot to research, but in this study, it was strongly affected by how matching was performed. This is not only a problem for research, since usability practitioners must match and merge problems for inspections with multiple evaluators, suggesting that matching and merging are unavoidable in usability work. Once multiple evaluators are used as recommended for HE (Nielsen 1994), the evaluators require analytical resources for matching and merging problems. These analysis resources may be procedural (e.g., for use by a chief evaluator), structural (based on matching report format elements), or social (e.g., merging by a group of evaluators in face-to-face discussions). Matching and merging resources determine the final problem set (types with instances) that forms the basis for understanding and addressing usability problems within iterative development. They are thus no less vital a resource than heuristics (criteria, dimensions), task selection (as in HW), or task walkthrough procedures (CW, HW, PW), and yet, along with problem reporting formats, they have received very limited attention from UIM developers and assessors.

56.4.4 Evaluation Resource Types for Usability Inspection

We can thus see that a wide range of resources will impact evaluators, with some resources being particularly relevant to specific steps in usability inspection, with which we can associate a *logical* (rather than temporal) structure with logically distinct phases that overlap during inspection. Inspection is essentially a search problem. Evaluator behavior is close to the generate and test strategy from early artificial intelligence research, where possibilities are first generated (e.g., moves in a chess game) and then tested to identify possibilities with specific attributes (e.g., the best next move). Note that the role of evaluation does not extend to fixing problems, even though methods such as *Rapid Iterative Testing and Evaluation* (RITE) (Medlock et al. 2005) have problem fixing within scope. RITE is structured around three questions: Is it a problem? Do we understand it?, and Can we fix it? Only the first question and some of the second are within scope of evaluation methods. The rest of RITE (really understanding and fixing problems) is an *iteration* method with (empirically) grounded redesign decisions, based on all relevant project information.

In the first *logical phase*, evaluators study UIMs and related resources. Also, each time a UIM is configured, augmented, and applied, evaluators must study the target design in the form provided, as well as the intended usage context. These *preparation* steps are followed by two logical generate and test phases of UIM application, which is followed by a final reporting phase. We can thus model usability inspection as having four distinct logical phases: (1) analyst preparation, (2) candidate problem discovery, (3) confirmation or elimination of candidate problems, and (4) problem reporting, with initial support for problem understanding and change recommendation. This results in four major *uses* of evaluation resources: preparation, discovery, analysis, and reporting. Evaluator preparation has been given limited attention in the UIM research literature. A notable exception is the preparation of tutorial material for HE (Lavery, Cockton and Atkinson 1996a), CD (Lavery, Cockton and Atkinson 1996b) and other UIMS. This HE tutorial was used in Cockton and Woolrych (2001).

Most UIMs focus on discovery and analysis. The DARe model (Discovery and Analysis Resources, initially DR-AR, Woolrych and Cockton 2002) was formed when exploring HE's scope and accuracy (Woolrych 2001), especially the causes of false positives. In any predictive method, analysts inspect a design for potential usability problems. Where multiple evaluators use a UIM, their problems must be merged into a "master" problem set. This will invariably contain false positives (problems predicted by evaluators that are not problems for actual users). Now, these can *only* be problems that have been *incorrectly analyzed*. They could not have been incorrectly discovered, as *the only discovery error is a genuine miss*—analysis is responsible for all other errors. Not all false positives are predicted by every evaluator, leading to a question: can some analysts correctly eliminate these through a process of analysis as true negatives? This conjecture arises directly from the DARe model, with its two logical phases, where analysts first discover possible usability problems, then through analysis either confirm them as probable problems or eliminate them as improbable problems.

The DARe model guided research to focus on the evaluation resources used in usability inspection and particularly those that were not provided by the UIM in use. Studies of problem reports have identified seven groups of knowledge resources that can influence problem discovery and analysis (Woolrych, Cockton, and Hindmarch 2005):

- **User** (knowledge of/beliefs about users, especially experience and abilities)
- **Task** (knowledge of what users want to do and how they prefer to accomplish this)
- **Domain** (domain knowledge that is specific to the system being evaluated)
- **Design** (knowledge and experience of interaction design principles or beliefs in lieu of)

- **Interaction** (knowledge of how humans actually interact with computers)
- **Technical** (knowledge of platform technologies such as browsers and toolkits)
- **Product** (information about the system and its capabilities)

These resources are known as *Distributed Cognitive Resources* (DCRs), so called because they are unevenly and unpredictably *distributed* across project contexts (UIMs, evaluators, other project resources). Evaluator effects (Hertzum and Jacobsen 2001) can be explained via different individuals possessing and/or accessing DCRs in different ways during inspection. All types of DCR have already been encountered above when reviewing UIMs, but this empirical study (Woolrych, Cockton, and Hindmarch 2005) showed all in use with HE, which provides few of them directly. In principle, MOT promotes *user* resources, as did early versions of CW, and PW supplies them directly. CW, HW, MOT, and PW require *task* resources. Nielsen's studies of HE usage have highlighted the importance of *domain* resources, as has Cockton and Woolrych (2001). HE and EC distill both *design* and *interaction* resources into heuristics and criteria, respectively. PW also provides both, as well as *product* and *technical* information, through its involvement of specialists. CW and CD embody theories of interaction. EC requires some product information to be provided. However, HE neither provides nor prompts for all seven groups of DCR, and yet Woolrych, Cockton, and Hindmarch (2005) were able to identify multiple instances of each in HE problem reports. Student evaluators were clearly augmenting HE with personal resources (they lacked a project context that could have provided any additional DCRs). Two evaluators in Cockton and Woolrych (2001) were asked to not use a specific UIM, but were instead given specific tasks to complete. Inspections that require no resources beyond the evaluator are called *expert inspections*. Here, expert inspections were supplemented with tasks based on domain goals (copying diagrams with a drawing tool), discovering problems that over 90 other student evaluators using HE did not.

DCRs can be used for both discovery and analysis. Extended structured problem reports that require evaluators to externalize their discovery and analysis steps are needed to distinguish the use of a DCR for discovery from use for analysis. System-centered resources (design, product, technical), as well as user-centered ones (user, task, domain, interaction), can guide discovery. The use of knowledge resources can be structured or unstructured, resulting in four broad classes of *problem discovery method*. There are system-centered unstructured and structured methods: *system scanning* and *system searching*. HE distills interaction design knowledge into heuristics. If HE in any way supports problem discovery, it is through heuristics that focus attention on design features such as error messages and navigation. Similarly, there are user-centered unstructured and structured discovery methods: *goal playing* and *procedure following*. These two discovery methods either use contextual information about users, tasks and domains or they substitute evaluators' beliefs about them. CW relies exclusively on task information to find problems. Appropriate task, technical, and domain knowledge can aid analysts in valid problem discovery and analysis. Knowledge of tasks allows analysts to discover problems that require sophisticated levels of interaction with the system before they become obvious. Simple system-centered discovery approaches would be unlikely to discover such problems.

Incorrect beliefs about users often lead to false positives, whereas product and interaction knowledge can avoid them (through correct elimination of improbable problems). Knowledge of how people interact with computers can imply that apparent design flaws will have no impact, for example, because users will never notice them (selective attention) or because they can learn what they need through interaction (distributed cognition). Technical and design knowledge can be used to correctly confirm a problem. However, reuse of resources in both discovery and analysis introduces risk of confirmation bias. Using the same resources in analysis as discovery adds nothing and is thus a source of false positives (Cockton, Woolrych, and Hindmarch 2004).

56.4.5 Project-Based Resources for Usability Inspection

Methods are only one aspect of usability work and thus successes or failures cannot wholly be explained in terms of resources within approaches. Instead, resources within the surrounding project context can have as much, if not more, influence on success or failure than the textbook resources of evaluation approaches in HCI.

We make no attempt here to survey all the factors that have been shown to shape usability work, but instead focus on some research that considered the broader context of method use. Theofanos and Quesenbery (2005), for instance, listed a variety of factors that influence how to report results, including the size of the recipient company, the kind of products evaluated, the audience, and the existence/absence of a formal usability process. Similarly, the software development approach used in a development project will influence usability work. Sy (2007) described how various approaches to usability evaluation had to be modified to fit the agile software development approach. For instance, reporting of problems was done through the daily scrum meetings and in process planning sessions, rather than through reports, and thus specific evaluation resources were not needed here (although usability roles may well have kept their own informal records).

We now quickly list several factors of which we are aware from research that has addressed how project- and organization-specific factors influence configuration and combination of evaluation resources. These include the following:

1. Client needs and expectations
2. Design purpose and vision
3. Project-, product-, and service-specific prioritization criteria
4. Business context
5. Budgetary and other logistical resources (e.g., time available)

6. Project leadership and design champions
7. Development approach and stage
8. Relation of usability work to the overall software development approach
9. Experience and competence of usability practitioners
10. Training and tutorial support on evaluation approaches
11. Professional/specialist education on general discovery and analysis resources in evaluation
12. Field research methods (users, tasks, etc.) and result communication formats
13. Task specification and notations used at design stage
14. Participant and evaluator recruitment strategies
15. Alignment of design purpose and evaluation purpose

The first five factors relate to the client and their economic/market context (or policy context for public sector projects). The next six factors (6–11) relate to the development team, with a specific focus on management and expertise. The last four factors (12–15) relate to design activities (last two focus on evaluation). These factors, and many others not listed above, severely limit the ecological validity of formal comparisons of evaluation methods, even in the unlikely event of confounds being so well managed that no issues could arise here.

Few factors above could be addressed substantially by generic HCI evaluation resources. All, however, are present to some degree in actual project and organizational contexts. Currently, the bulk of the resources that are critical to working out methods within usability work are external to documented HCI evaluation approaches. Given this, academic attempts to "compare methods" are bound to "fail the practitioner" (Wixon 2003), since as far as providing comprehensive support for usability work is concerned, all current documented approaches have already failed practitioners to some extent by not providing them with comprehensive resources. However, much of the success of "discount methods" such as HE may be due to the sparseness and flexibility of the provided resources, which let skilled evaluators fill gaps with resources from personal, project, or organizational sources.

56.5 ASSESSING AND COMPARING USABILITY INSPECTION METHODS

Having identified the broad range of evaluation resources that are needed, but with few provided directly by UIMs, it should be clear that assessing and comparing UIMs could be a hopeless endeavor, since what actually gets used is *never the UIM alone* (at least for current UIMs), since evaluators must draw on DCRs beyond a UIM to be able to carry out usability work. However, it is still worth reviewing the two main broad approaches for assessing and comparing UIMs—one qualitative and the other quantitative. *Qualitative* approaches code predictions as a basis for scoping the capabilities and defects of UIMs. *Quantitative* approaches relate the performance of one method to another, often expressed as a percentage, for example, to express proportions of correctly predicted problems. We first review quantitative approaches and then review qualitative ones. The critiques of both approaches to assessing UIMs as complete wholes adds to the insurmountable challenges for direct assessment and scoping of "pure" UIMs, which are next summarized. With this critique in place, we review how resources beyond the assessed and compared UIMs can account for differences in evaluator performance as readily, if not more so, than the few evaluation resources that a UIM actually provides.

56.5.1 Summative Assessment and Comparison

The fundamental purpose of usability inspection is to find and report any potential user difficulties for remedy in redevelopment. Ideally, a UIM would find *all* usability problems that may arise from use of an interactive system; in other words, it would be *thorough* and, hence, not miss any elements that may cause user difficulties. The final problem set achieved through usability inspection would also only comprise *valid* problem predictions, with no false alarms reported. Consequently, finding all problems without false alarms, the result of a usability inspection would be wholly effective.

Unfortunately, any predictive methods such UIMs are not completely thorough nor are the predictions always valid with the subsequent impact on UIM effectiveness. *Validity*, *thoroughness*, and *effectiveness* are quantitative measures of UIM capability for which Sears (1997) suggested the following formulae:

$$\text{Validity} = \frac{\text{number of } \textit{real} \text{ usability problems } \textit{found} \text{ by UIM}}{\text{number of usability problems } \textit{predicted} \text{ by UIM}}$$

$$\text{Thoroughness} = \frac{\text{number of } \textit{real} \text{ usability problems } \textit{found} \text{ by UIM}}{\text{number of } \textit{real} \text{ problems that } \textit{exist}}$$

$$\text{Effectiveness} = \text{Validity} \times \text{Thoroughness}$$

The scores derived from the formula are between 0 and 1, where 1 would be the *perfect* score. Such scores are often expressed as percentages (e.g., 0.5 = 50%). For validity, a perfect score (of 1) would mean *all* problems found by the UIM would be real problems, that is, with no false alarms. Similarly, if the count of real problems found by the UIM equals the count of problems that exist, this too would result in a perfect thoroughness score of 1. With perfect thoroughness and validity, effectiveness too would also be a perfect 1 ($1 \times 1 = 1$), that is, *all predictions valid, no false alarms, all problems found*. With thoroughness of 0.7 and validity of 0.5, effectiveness would be 0.35.

Many assessments of UIMs have only used thoroughness as a quantitative measure and have shown, for example, how the use of multiple evaluators increases thoroughness (Nielsen and Landauer 1993). However, when validity is also calculated, the benefits of rising thoroughness are soon undermined by falling validity and hence reduced effectiveness (Cockton and Woolrych 2002). Giving equal weight to thoroughness and validity may not be suitable for all evaluation contexts (Hartson, Andre, and Williges 2001). However, relying solely on thoroughness and validity measures is inadvisable: as well as it being extremely difficult to calculate validity, they have no diagnostic power for explaining a lack of thoroughness or validity, and thus cannot guide the selection of complementary methods that can compensate for a UIM's weaknesses.

Two values in Sears' equations present severe research challenges. First, the denominator for thoroughness (number of real problems that exist) relies on the implicit *closure* of "all of the usability problems for a design," but "all" can never be determined confidently: there can always be undiscovered problems. All thoroughness measures are always upper bounds that reduce as newly discovered real problems increase the denominator. For the validity numerator (number of real usability problems found by UIM), we only need to find real usability problems that match those predicted by a UIM. Hence all validity measures are always lower bounds and increase if any newly discovered real problems match predictions, increasing the numerator.

While Sears' formulae are attractive and offer the simplest possible way of comparing UIMs, they rely on closures that mostly exist in mathematics (closure of a set under some operation). In the real world, we use the word *all* for *enumerations*, such as "prohibited items in carry-on baggage," where closure results from human decisions, and even this uses very broad categories (e.g., ammunition; automatic weapons; axes and hatchets, unless part of aircraft equipment; billiard cues; billy clubs; and blackjacks). However, *all* in the context of thoroughness has the sense of a closure and not an arbitrary human enumeration. Such conceptual dead ends are compounded by miscounts of problems that are discovered through user testing (the default source of number of *real* problems that *exist*). Evaluators appear to miss most problems in video data (Jacobsen, Hertzum, and John 1998), and further *miscounts* may arise from poor merging of usability problem sets (Cockton and Lavery 1999). Connell and Hammond (1999) showed that the impact of multiple evaluators on cumulative problem counts depends on how predicted problems are merged, resulting in a slower increase in cumulative thoroughness than showed in Nielsen and Landauer (1993). Also, specificity of description determines problem count according to Cockton and Lavery (1999). As usability problems become more specific, problem counts rise. For example, "misleading status bar messages" is one problem, but "incomplete prompts" and "technical vocabulary in message" are two, and more would result if we distinguished between prompt problems such as "no instruction to close free form with double click" or "no instruction to use control-click to delete vertices" (for drawing objects).

For thoroughness, *asymptotic testing* has been proposed as a means to ensure the highest possible denominator (Hartson, Andre, and Williges 2001). However, this is generally understood as continuing to test additional users using the same test protocol until no new problems emerge. Changing the test protocol, such as between different sets of fixed tasks, between free and fixed tasks, or between individual think aloud and peer tutoring, also reveals new problems (Lindgaard and Chattratichart 2007). Also, adding analysts and/or applying a structured method such as SUPEX could extract more problems from video data. As with most attempts at closure, methods to find *all* problems simply expose a further unachievable closure, that is, in *all of the ways to find all* the usability problems. However, maximizing the numerator for *validity* is more tractable. When real problems are found by fixed task testing, a *falsification methodology* can expose users repeatedly to predicted problems (Woolrych, Cockton, and Hindmarch 2004). Missed problems here will have two types of cause: errors in test task planning (which can be fixed) and an inability to put the system in a state where the problem would appear (which can be very difficult to achieve; Lavery and Cockton 1996).

Some quantitative measures are not prone to insurmountable mixes of conceptual and practical impossibilities. Cockton and Woolrych (2001) coded heuristic applications for nonbogus predictions as being appropriate or inappropriate, based on explicit criteria from the HE training manual (Lavery, Cockton, and Atkinson 1996a) provided for analysts. This provided conformance questions that stated, "What the system should do, or users should be able to do, to satisfy the heuristic" (p. 4). Such questions are answered with conformance evidence, such as the "design features or lack of design features that indicate partial satisfaction or breaches of the heuristic" (p. 4). For many heuristic applications, these criteria were clearly ignored. Only 39% of the heuristics applied to nonbogus predictions were appropriate (and only 31% for successful predictions). Such measures of appropriateness are important in identifying the extent to which problem discovery and analysis are due to UIM or other evaluation resources. Heuristics tended to be best applied to predictions that turned out to be of low frequency and/or severity. These were likely to be predictions that seemed less probable than ones that turned out to be of high frequency and/or severity, which presumably were so obvious that the 13 heuristics (Nielsen 1994a, plus three specific to visualization; Lavery, Cockton, and Atkinson 1996a) were not properly reviewed to strengthen analysis. Furthermore, given HE's known limited coverage, there may be no appropriate heuristic for some predictions. Nielsen (1994b) found that seven factors could only account for 30% of the variability in his corpus of 249 usability problems, and 53 factors were needed to account for 90%, too many for HE. The names of the seven factors, which were chosen by Nielsen, included "visibility of system status," "match between system and real world," and "error prevention." This supported his claim that the factors (the first seven heuristics in Figure. 56.1 along with 8 and 9) form a basis for a set of heuristics that has remained the main resource for HE; the 10th, "Help and Documentation," was later added. It is not clear, however, how much the additional three heuristics extend the coverage of the sample problem set beyond 30%, but this is very unlikely to reach 40%. As a result, HE cannot cover most likely usability problems, and the ones it does cover are not always high frequency ones.

56.5.2 Qualitative Approaches to Coding Usability Problems

Quantitative measures can be seductive, so it is important to approach them critically. Even if simple scores for thoroughness or validity could be reliably calculated (and to be clear, they cannot), identical scores could hide major differences in UIM performance (Woolrych and Cockton 2000). For example, two UIMs could have identical thoroughness scores, but one could systematically miss most severe problems, which has been shown to be true for HE (Cockton and Woolrych 2001) by *coding* successful predictions and missed problems for *discoverability*. The easiest problems to discover are *perceivable*: these can be seen at a glance. Next comes *actionable* problems: these can be discovered after a few simple actions (e.g., a mouse click). Hardest to find are *constructable* problems, which require complex task scenarios to reveal them. 80% of problems missed by evaluators using HE were constructable, as opposed to 7% of successful predictions. HE's limited use of task knowledge resources is responsible for missing usability problems that only arise in interaction sequences longer than a few simple actions.

Discoverability is a *qualitative* measure, requiring UIM assessors to code a problem as perceivable, actionable, or constructable. Useful codes delve below the surface of superficial quantitative measures to reveal the impact of a UIM's (lack of) evaluation resources. However, the ability to code for a specific measure depends on the contents of the reporting formats for usability problems. In an experiment to establish common ground for method assessment, Cockton and Woolrych (2009) analyzed eight multilingual usability report problem sets from various evaluation methods (HE, EC, user testing, expert inspection) in a variety of formats (e.g., research problem set, consultancy report). The problem reports were inspected for common elements that could be a basis for comparison through problem coding, including the following:

- Description of the usability problem
- The context in which the problem was discovered
- Evidence of how the usability problem was discovered
- Actual or predicted user difficulties relevant to the problem
- Evidence of the frequency of the problem
- Any suggested solutions to the problem

The only common element in all of the problem report sets was a problem description! The context of problem discovery was quite common in most (but not all) problem sets, but often within a problem description. Thus comparing UIMs on the basis of qualitative measures requires appropriate problem reporting formats. Identification of DCRs in research by the first two authors, and analysis of interactions between them, depended on the ability to code problem reports for the use of a range of evaluation resources and the ability to distinguish their separate use on discovery and analysis, which was not always possible. Severity codings were used in early studies of HE by the first two authors to reveal the interaction between appropriateness of heuristic use of the impact of actual problems. This suggested that appropriate heuristic use was a form of confirmation bias arising when considering predictions that turned out to have low frequency and/or severity in the set of actual problems revealed through falsification testing (Cockton and Woolrych 2001).

56.5.3 Why Comparisons of Usability Inspection Methods Fail to Produce Reliable Results

Initially, problems in comparing UIMs were attributed to poor experimental design. Gray and Salzman (1998) reviewed the validity of five major studies in the literature that compared the performance of different evaluation methods. They based their review on Cook and Campbell's (1979) four main types of validity: statistical conclusion validity, internal validity, construct validity, and external validity. They added a fifth type: conclusion validity. Gray and Salzman found validity problems with each study reviewed. For example, Jeffries et al. (1991) compared CW, expert inspection (but called HE), guidelines, and user testing and found expert inspection to be superior. However, the study used few evaluators and thus it is not clear if the effects of the study (e.g., the superiority of expert review) occurred by chance and lacked *statistical conclusion validity.*

Internal validity is threatened when the effect supposedly caused by a variable under manipulation is actually caused by a confounding variable. One study where expert review was shown to be superior may, however, suffer from internal validity because the experts were given a 2-week period at their own pace to complete the evaluation, whereas evaluators in the other conditions had far less time. The worst performing methods were used by software engineers with less usability expertise than the HE group. Here, *time* is revealed as an evaluation resource as well as additional DCRs that usability experts bring to evaluations.

Nielsen (1992) investigated the effects of evaluator expertise on the number of problems identified using HE and claims that "usability specialists were much better than those without usability expertise by finding usability problems with heuristic evaluation." Gray and Salzman claim this study lacks *conclusion validity* because it is not clear what the effect of HE was on the evaluator's ability to find usability problems or how many problems evaluators would find through expert review without recourse to HE's evaluation resources. They suggest the data supported the more modest claim that "experts named more problems than non-experts."

Gray and Salzman found that many studies presented conclusions that were not supported by the data. They do not argue against presenting advice based on experience rather than experimental evidence, but argue that the source of such advice should be made explicit. Since their critique, our knowledge of the range of evaluation resources that influences evaluator performance has significantly increased, both through attempts to avoid the methodological flaws identified by Gray and Salzman and by more recent realistic case studies of usability work. The former research tactics (avoiding methodological flaws) have exposed the impact of problem reporting formats, matching and merging procedures, and task selection resources on UIM performance. As only HW provides any such resources (task selection), no UIM can be assessed in isolation. It must be supplemented by other evaluation resources, which may themselves have a major impact on evaluator performance (as in Cockton et al. 2003). Studies of usability work have revealed an even wider range of evaluation resources that have far more impact on evaluation outcomes than resources provided by UIMs. Usability work requires evaluation resources that are rarely provided by UIMs, which thus cannot be compared by comparing the outcomes of evaluations, since these will be confounded by many evaluation resources in the study context. However, it is possible to manipulate a single resource such as a problem report format and hold other evaluation resources fairly constant (through randomization). Such focused studies, rather than crude comparisons of UIMs, are one important way forward in HCI research on evaluation methods. The methodological challenges identified by Gray and Salzman simply cannot be overcome, but there have been some useful spin offs from continued attempts to rigorously compare UIMs.

56.6 CHOOSING AND USING USABILITY INSPECTION METHODS

Based on the discussion so far, it should be clear that choosing, combining, and using UIMs are creative processes; simply picking an inspection method because it is easy to use or applicable early in a project is insufficient and likely to fail. Next we provide some guidance on planning usability inspections, but temper this with an overview of the need to select resources that can (and will) be provided by a specific project.

56.6.1 Planning Usability Inspections

The DCRs identified as a result of exploring the DARe model do not simply enumerate the groups of resources used in usability inspection; DCRs are also a checklist for things to think through when preparing for an inspection. For instance, many usability inspections fail because evaluators know too little about the product, users' tasks, and the domain to be supported. Many studies have documented how knowledge about these things may improve inspection, to the extent that some evaluation approaches successfully trade in the need for knowledge about interaction and technical issues for domain knowledge (e.g., Følstad 2007). In planning inspections, usability evaluators need to consider whether knowledge resources concerning products, users' tasks, and the domain are available and sufficient for the inspection. Ideally, project managers and usability specialists need to plan support for inspections from the project outset to ensure that the necessary resources are in place to support high quality inspection. Such resources have multiple uses beyond supporting usability inspection. They also support planning of participant recruitment and selection of tasks (when appropriate) for user testing and also design activities such as authoring personas and scenarios. Planning for usability inspections should not start just before inspections are scheduled. Instead, resources required for all user-centered design activities need to be identified early in a project and sourced in good time. Otherwise, it is unreasonable to expect usability inspection to have high *downstream utility* (John and Marks 1997), which would be achieved by correcting, identifying, and understanding problems and recommending appropriate design change recommendations on this basis. Downstream utility cannot compensate for *upstream futility;* if evaluators are not adequately supported, then poor quality inspections result. In this sense, the tactics underlying *discount methods* are misguided. While a UIM can be reduced to 10 heuristics, advice on using multiple evaluators and a very rough two pass procedure, the effectiveness of UIMs such as HE is substantially compromised by deficiencies across all DCRs.

The DCRs revealed via the DARe model indicate that many different resources enter usability inspection. This is in agreement with the recommendation that multiple analysts may improve inspection method thoroughness (Nielsen and Landauer 1993). The DARe model explains why multiple evaluators find more problems when the UIM and environment remain the same. Quite simply, as more evaluators are added, more problem discovery resources are added to the inspection, hence more problems are found. However, the DARe model also leads us to considering the impact of multiple evaluators on analysis resources. If evaluators are inclined to not reject problems at all or if they are unduly confident in confirming problems, then multiple evaluators will not only be collectively more thorough, and thus find more problems, but they will also be collectively less valid and thus predict more false positives (Cockton and Woolrych 2002).

The more recent focus on *usability work* has greatly extended DCRs with a broader understanding of project-specific resources, as discussed earlier. These too provide another checklist for planning evaluation support within software and hardware development projects. For instance, the *goal* of the evaluation is a consideration too frequently stepped over in planning. The goal may be a formative or a summative evaluation; a global evaluation or a targeted assessment of a feature; negotiated among stakeholders or relatively open; and to identify solutions or pinpoint difficulties. Being explicit about the goal and configuring usability work to reflect it is crucial: it especially supports the logical phases of confirmation or elimination of candidate problems and problem reporting. Furthermore, the goals of evaluation need to focus on design purpose (Cockton 2005). This can be encouraged by report formats that include a focus on business impact or other forms of design purpose (e.g., social or personal impact), as illustrated by Hornbæk and Frøkjær (2008b). As already noted, reporting formats and merging and matching procedures should also be considered. Again, when planning an inspection, considering these and other resources would provide useful input to planning. Throwing evaluation resources together in a rush is a recipe for ineffective poor quality inspections

Planning for usability inspection is thus best considered as one aspect of overall project management. While usability inspections can be "slipped in" at short notice into a project schedule and will always deliver some value, it is better to plan for them within an overall user-focused process, balancing other concerns such as business impact (e.g., for e-commerce sites) as appropriate. DCRs, including project-specific resources identified above, provide a checklist that can support planning. Advance planning allows time for careful consideration of heuristic sets, with project-specific amendments and extensions as necessary. It also allows time for careful consideration of severity scales, which must be closely related to design purpose. The severity of a usability problem is directly related to its impact on achieving the intended purpose for an interactive system or device. Severity problems increase as usability problems progressively degrade the value or worth that either users can gain from interaction or the achievable worth for other stakeholders such as website owners or other project. Without project-specific severity scales in place, evaluators must use *ad hoc* generic scales that are close to meaningless.

56.6.2 Matching Approach Resources to Project Resources

Project planning and resourcing will always be constrained by a range of internal and external considerations. For example, client needs and expectations will control what can and cannot be considered as evaluation resources. The first two authors have worked on a range of usability consultancies where the client insisted on exclusive use of user testing, even where it was clear from a brief "guerilla" inspection that much better value would arise for user testing if an inspection was followed by a set of design changes first. However, many clients find inspection methods too subjective and unscientific, even when evaluators can offer decades of combined usability expertise. This mirrors trends from surveys of usability professionals. HE was the most used usability method in a paper questionnaire-based survey of 111 usability professionals in 1999 (Rosenbaum, Rohn, and Humburg 2000). However, a more recent web survey of 83 usability professionals, with a detailed follow up for 16 respondents (Venturi, Troost, and Jokela 2006), now ranks inspection methods (38% of respondents use during design) well below discount user testing (quick and dirty usability tests used by 53%—in contrast to 70% and 65%, respectively, in the 1999 survey). Although surveys must be compared with great caution, UIMs may no longer be the predominant method in usability practice. User-based methods are increasingly preferred because, once selected and briefed, individual test participants generate many complete usage scenarios. Hence, user testing can expose problems through user behaviors that are very hard to anticipate. However, testing only reveals a wide possible range of usability problems if users stress the system with complex interactions across all features. It is thus possible to anticipate user difficulties that may not emerge during testing, unless specific efforts are made to flush out all predicted problems (Woolrych, Cockton, and Hindmarch 2004). However, UIMs are still in widespread use, sometimes because the cost and logistical challenges of recruiting test participants leads some clients to rule out user testing, placing the whole usability evaluation load on UIMs (including expert inspection).

One of the most important resources for evaluation is a clear sense of design purpose and project vision. Such a resource can be frustratingly rare, but without it makes it very difficult to devise adequate severity scales, nor is there a sound basis for prioritizing design changes on the basis of well grounded criteria. For commercial systems, it is important to relate purpose and priorities to the business context. Without such resources, there is limited support for considering business impact during usability inspections. As with other key evaluation resources, in the absence of information and/or adequate project leadership, usability specialists need to find ways to compensate for lack of support from project management. Even when there is clear vision and good project leadership, not all recommended resources for UIMs can be provided. For example, multiple evaluators have been shown to improve thoroughness for HE and other UIMs, but these evaluators have to be sourced, and finding 5–8 expert evaluators with HCI expertise will not be possible for most projects. However, usability leads may be able to provide some training in UIMs for other development staff. Alternatively, a Pluralistic Walkthrough (PW) can be chosen, with specialists with roles such as marketing, training, accessibility, support, or documentation providing alternative complementary expertise to the usability specialist.

Inspections need to take place as soon as any aspect of a design can be evaluated. UIMs provide their best value early in the development cycle. The longer design choices are adhered to, the harder it is to make changes as a result of an inspection, which is sometimes why clients require user testing, since more credible but more costly data may be needed to establish the need to undo existing expensively implemented design decisions. A common exception here are those web sites (e.g., e-commerce websites) that are often redesigned at set intervals, with a corresponding expectation that most existing design decisions can be revisited and revised if necessary.

56.7 FUTURE OF USABILITY INSPECTIONS: MAKING DIFFERENCES TO USABILITY WORK

UIMs are approaches that offer bundles of resources that can be adapted and complemented to support usability work. Thus, while there is much evidence of difficulties in UIM use, especially as regards evaluation quality, most problems associated with the use of UIM resources such as heuristics and walkthrough procedures are avoidable. Our view remains that, used with care, UIMs remain key approaches in the usability toolbox: "The challenge is to improve all HCI methods, so that discount methods are less discounted and 'full strength' methods can be applied in more contexts" (Cockton and Woolrych 2002, p. 29). The most appropriate use of UIMs is to drive design iterations, rather than use for summative evaluation, benchmarking, or competitor analysis. UIMs are cheap to apply and are seen as low cost and low skill (but more resources and expertise can greatly improve evaluation performance). They have been applied to many commercial designs by practitioners. UIMs can be used before a testable prototype has been implemented and can be iterated without exhausting or biasing a group of test participants. UIMs can also be a planning resource for user testing, which can be designed to focus on predicted problems. Development resources may rule out user testing, leaving UIMs as the only possible approach, and some usability is always better than no usability.

UIMs remain important and their weaknesses can be mitigated. The main risks are missing serious usability problems and wasting development effort through poor fixes to mispredicted nonproblems (false positives). Different business models make different demands; thus, UIM errors are more costly in some development contexts than others are. In some contexts, successful predictions may always be worthwhile despite a flood of false positives, as fixed problems translate into savings on support costs and attractive new features. When users buy software, most must struggle on, unlike visitors to free e-commerce sites, yet mainstream UIMs do not cover their complete user experience, especially for affective issues such as trust, comfort, and brand image. Hence, user testing is vital to eliminate any severe problems; otherwise, money will be lost.

Novice evaluators should learn how to configure and augment UIMs before practice on familiar systems and usage contexts to establish quickly their scopes and accuracies. No UIM provides contextual information or understandings of users and tasks (Cockton and Woolrych 2002). Fortunately, HCI professionals apply contextual research (Venturi, Troost, and Jokela 2006) even though some may not accept that contextual resources are not part of UIMs (Manning 2002). Novice UIM users need to be coached to properly augment UIM resources with an appropriate complement of DCRs and make full use of available project resources. In particular, evaluators must know about and understand the system they are inspecting. Some UIM developers disagree, preferring to keep analysts in untainted ignorance. They aim to induce user empathy, but this can only be truly grounded in contextual resources. Ungrounded beliefs about users and usage contexts can be very unreliable (and often insult real users' intelligence). If evaluators begin inspection ignorant of both the usage context and the system being evaluated, then the result is not better empathy with the user but incorrect claims that features are missing or beyond users. While there may be evidence of usability problems in such evaluator errors, they are still errors. Properly informed evaluators are more likely to note that a feature is hard to find (rather than absent) or that a design rationale has overlooked some contextual criteria. Such problem predictions are far preferable to bogus ones, which are major risks in UIM usage. In short, evaluator preparation is crucial to successful UIM usage.

By properly configuring and augmenting UIMs, evaluators can improve the quality of discovery and analysis resources. Knowledge of expected contexts of use, human capabilities, and key properties of HCI are critical. For example, knowledge of display-based interaction may eliminate possible problems that overlook users' abilities to discover information and to explore interactive behaviors. Knowledge of human capabilities such as visual attention may either confirm possible problems (e.g., key information in the wrong place) or eliminate them (e.g., misleading information in the wrong place).

Evaluators thus need to be expected to work to make UIMs work. However, in the next few years, we expect to see a growth of in-depth case studies of evaluation practices and outcomes in the HCI literature. Usability specialists will be able to draw on these case studies to improve their practice. Note here that the aim of HCI research on UIMs is moving away from method assessment and comparison towards studies of usability work. The aim is less to improve UIMs per se, but instead to improve their usage and to do so by improving evaluators through improved intelligence on the inspection process. At some point, a critical mass of well designed and well executed case studies will support derivation of complex system models of usability work, from which hypotheses can be derived. Well focused experimental studies can then be designed to test these hypotheses. At this point, the scientific knowledge sought in initial comparisons of UIMs will be more likely, but it will not take the form of simple rankings of UIMs, but instead deep understandings of the impact on evaluation performance of specific configurations and combinations of evaluation resources. Until then, expertise in usability inspection is primarily the responsibility of evaluators. UIMs themselves cannot be sufficiently improved to guarantee success. However, imaginative use of specific resources such as report formats and specialized heuristics can have disproportionate impacts. The core knowledge for usability inspection does not, thus, concern "methods" but resources. Evaluators need to understand the bases for quality for each evaluation resource and put this knowledge to the best effect when configuring and combining resources for usability inspection.

We expect the next few years to open up new research approaches for UIMs (Woolrych, Hornbæk, Frøkjær and Cockton 2012), which will feed through into education and practice. These new approaches will firstly focus within methods on different types of resource (e.g., knowledge, scoping, procedural, analytic) and the impact of variations for a single resource, and secondly above the method level on usability *work activities*. Methods are simply the wrong unit of analysis for understanding evaluation practice. By focusing on methods, research has not paid enough attention to variations within use of the same method, and has thus attributed specific impacts to methods that are actually impacts of specific resources. The impact of different formats and content for problem report formats has clearly demonstrated that specific resources can have disproportionate impacts on evaluation performance. Also, by focusing on individual evaluation methods, research has not paid enough attention to how sets of methods are selected, adapted and combined during usability work. What matters in usability work are the outcomes of specialist investigations and advice, and that is the result of complex interactions within a mix of contextual factors. Methods, or rather the bundling of resources into approaches, are only aspect of this mix. Methods have to be related to the big picture in interaction design work. We expect that this big picture will be a major focus of HCI research in the current decade, resulting in a much more sophisticated understanding of usability work and the role of methods within this.

REFERENCES

Andre, T. S., H. R. Hartson, S. M. Belz, and F. A. McCreary. 2001. The user action framework: A reliable foundation for usability engineering support tools. *Int J Hum Comput Stud* 54(1):107–36.

Bastien, J. M. C., and D. L. Scapin. 1995. "Evaluating a user interface with ergonomic criteria." *Int J Hum Comput Interact* 7(2):105–21.

Bias, R. G. 1994. The pluralistic usability walkthrough: Coordinated empathies. In *Usability Inspection Methods*, ed. J. Nielsen and R. L. Mack. New York: John Wiley and Sons.

Capra, M., and T. Smith-Jackson. 2005. *Developing Guidelines for Describing Usability Problems (No. ACE/HCI-2005-002)*. Blacksburg, VA: Virginia Tech, Assessment and Cognitive Ergonomics Laboratory & Human–Computer Interaction Laboratory.

Cockton, G. 2005. A development framework for value-centred design. In *CHI'05 Extended Abstracts on Human Factors in Computing Systems*, ed. G. C. van der Veer and C. Gale, 1292–195. New York: ACM Press.

Cockton, G., and D. Lavery. 1999. A framework for usability problem extraction. In *Proceedings of Interact '99*, ed. M. A. Sasse, and C. Johnson, 344–52. Amsterdam, The Netherlands: IOS Press.

Cockton, G., and A. Woolrych. 2001. Understanding inspection methods: Lessons from an assessment of heuristic evaluation. In *People and Computers XV: Interaction without Frontiers*, eds. A. Blandford and J. Vanderdonckt, 171–91. London: Springer-Verlag.

Cockton, G., and A. Woolrych. 2002. Sale must end: Should discount methods be cleared off HCI's shelves? *Interactions* 9(5):13–8.

Cockton, G., and A. Woolrych. 2009. *Comparing UEMs: Strategies and Implementation*. Final report of COST-294 working group 2. http://141.115.28.2/cost294/upload/533.pdf. (Accessed November 30, 2011).

Cockton, G., A. Woolrych, L. Hall, and M. Hindmarch. 2003. Changing analysts' tunes: The surprising impact of a new instrument for usability inspection method assessment. In *People and Computers XVII: Designing for Society*, eds. P. Palanque, P. Johnson, and E. O'Neill, 145–62. London: Springer-Verlag. John Long Award for Best Paper.

Cockton, G., A. Woolrych, and M. Hindmarch. 2004. Reconditioned merchandise: Extended structured report formats in usability inspection. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, ed. E. Dykstra-Erickson and M. Tscheligi, 1433–6. New York: ACM Press.

Connell, I. W., and N. V. Hammond. 1999. Comparing usability evaluation principles with heuristics: Problem instances vs. problem types. In *IFIP INTERACT '99: Human–Computer Interaction*, eds. M. A. Sasse and C. Johnson, 621–9. Amsterdam, The Netherlands: IOS Press.

Cook, T. D., and D. T. Campbell. 1979. *Quasi-Experimentation: Design and Analysis Issues for Field Settings*. Chicago: Rand McNally.

Cuomo, D. L., and C. D. Bowen. 1992. Stages of user activity model as a basis for user-system interface evaluations. In *Proceedings of the Human Factors Society 36th Annual Meeting, Human Factors Society*, 1254–8. Santa Monica, CA: Human Factors Society.

Cuomo, D. L., and C. D. Bowen. 1994. Understanding usability issues addressed by three user-system interface evaluation techniques. *Interact Comput* 6(1):86–108.

Frøkjær, E., and K. Hornbæk. 2002. Metaphors of human thinking in HCI: Habit, stream of thought, awareness, utterance, and knowing. In *Proceedings of HF2002/OzCHI 2002 (CD-Rom)*, eds. R. Kuchinsky, L. Johnson, and F. Vetere. Australia: CHISIG.

Frøkjær, E., and K. Hornbæk. 2008. Metaphors of human thinking for usability inspection and design. *ACM Trans Comput Hum Interact* 14(4):1–33.

Furniss, D. 2008. *Beyond Problem Identification: Valuing Methods in a 'System of Usability Practice.'* London: University College London.

Følstad, A. 2007. Work-domain experts as evaluators: Usability inspection of domain-specific work-support systems. *Int J Hum Comput Interact* 22(3):217–45.

Gray, W. D., and M. Salzman. 1998. Damaged merchandise? A review of experiments that compare usability evaluation methods. *Hum Comput Interact* 13(3):203–61.

Green, T. R. G. 1991. Describing information artifacts with cognitive dimensions and structure maps. In *Proceedings of the HCI'91 Conference on People and Computers VI*, eds. D. Diaper and N. Hammond, 297–315. Cambridge, UK: Cambridge University Press.

Green, T. R. G., and D. Benyon. 1995. Displays as data structures: Entity-relationship models of information artifacts. In *Proceedings of INTERACT'95: IFIP TC13 Fifth International Conference on Human–Computer Interaction*, eds. K. Nordby, P. Helmersen, D. Gilmore, and S. Arnesen, 55–60. London: Chapman & Hall.

Green, T. R. G., and D. Benyon. 1996. The skull beneath the skin: Entity-relationship models of information artefacts. *Int J Hum Comput Stud* 44(6):801–28.

Green, T. R. G., and M. Petre. 1996. Usability analysis of visual programming environments: A 'cognitive dimensions' framework. *J Visual Lang Comput* 7(2):131–74.

Hartson, H. R., T. S. Andre, and R. C. Williges. 2001. Criteria for evaluating usability evaluation methods. *Int J Hum Comput Interact* 13(4):373–410.

Hertzum, M., and N. E. Jacobsen. 2001. The evaluator effect: A chilling fact about usability evaluation methods. *Int J Hum Comput Interact* 13(1):421–43.

Hornbæk, K., and E. Frøkjær. 2002. Evaluating user interfaces with metaphors of human thinking. In *Proceedings of User Interfaces For All*, Paris, France, October 23–25, also in Springer Lecture Notes in Computer Science, Vol. 2615, 486–507. Berlin: Springer.

Hornbæk, K., and E. Frøkjær. 2004. Usability inspection by metaphors of human thinking compared to heuristic evaluation. *Int J Hum Comput Interact* 17(3):357–74.

Hornbæk, K., and E. Frøkjær. 2006. What kinds of usability-problem description are useful to developers? In *Human Factors and Ergonomic Society's Annual Meeting*, 2523–7. Santa Monica, CA: Human Factors Society.

Hornbæk, K., and E. Frøkjær. 2008a. Comparison of techniques for matching of usability problem descriptions. *Interact Comput* 20(6):505–14.

Hornbæk, K., and E. Frøkjær. 2008b. Making use of business goals in usability evaluation: An experiment with novice evaluators. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, 903–12. New York: ACM Press.

Howarth, J., T. S. Andre, and R. Hartson. 2007. A structured process for transforming usability data into usability information. *J Usability Stud* 3(1):7–23.

Hvannberg, E. T., E. L.-C. Law, and M. K. Lárusdóttir. 2007. Heuristic evaluation: Comparing ways of finding and reporting usability problems. *Interact Comput* 19(2):225–40.

Jacobsen, N. E., M. Hertzum, and B. E. John. 1998. The evaluator effect in usability tests. In *Human Factors in Computing Systems CHI'98 Summary*, eds. C.-M. Karat and A. Lund, 255–6. New York: ACM Press.

Jeffries, R. 1994. Usability problem reports: Helping evaluators communicate effectively with developers. In *Usability Inspection Methods*, eds. J. Nielsen and R. L. Mack, 273–94. New York: John Wiley and Sons.

Jeffries, R., J. R. Miller, C. Wharton, and K. M. Uyeda. 1991. User interface evaluation in the real world: A comparison of four techniques. In *Proc. CHI'91 Conf. on Human Factors in Computing Systems*, eds. S. P. Robertson, G. M. Olson, and J. S. Olson, 119–24. New York: ACM.

John, B. E., and S. J. Marks. 1997. Tracking the effectiveness of usability evaluation methods. *Behav Inf Technol* 16(4/5):188–202.

John, B. E., and H. Packer. 1995. Learning and using the cognitive walkthrough method: A case study approach. In *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, eds. I. Katz, R. Mack, and L. Marks, 429–36. New York: ACM Press.

Lavery, D., and G. Cockton. 1996. Iterative development of early usability evaluation methods for software visualisations. In *Proceedings of the 6th Workshop of Empirical Studies of Programmers*, eds. W. D. Gray and D. A. Boehm-Davis, 275–6. Ablex. Glasgow, UK: Glasgow University.

Lavery, D., G. Cockton, and M. Atkinson. 1996a. *Heuristic Evaluation: Usability Evaluation Materials*. Technical Report TR-1996-15, Department of Computing Science. Glasgow, UK: University of Glasgow.

Lavery, D., G. Cockton, and M. Atkinson. 1996b. *Cognitive Dimensions: Usability Evaluation Materials*. Technical Report TR-1996-17, Department of Computing Science. Glasgow, UK: University of Glasgow.

Lavery, D., G. Cockton, and M. P. Atkinson. 1997. Comparison of evaluation methods using structured usability problem reports. *Behav Inf Technol* 16(4):246–66.

Lewis, C., P. Polson, C. Wharton, and J. Rieman. 1990. Testing a walkthrough methodology for theory-based design of walk-up—and-use interfaces. In *Proc. CHI'90 Conf. on Human Factors in Computing Systems*, eds. J. Carrasco and J. Whiteside, 235–42. New York: ACM Press.

Lewis, C., and C. Wharton. 1997. Cognitive walkthroughs. In *Handbook of Human–Computer Interaction*, 2nd ed., eds. M. Helander, T. K. Landauer, and P. Prabhu, 717–32. New York: Elsevier.

Lindgaard, G., and J. Chattratichart. 2007. Usability testing: What have we overlooked? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1415–24. New York: ACM Press.

Mankoff, J., A. K. Dey, G. Hsieh, J. Kientz, M. Ames, and S. Lederer. 2003. Heuristic evaluation of ambient displays. CHI Letters, CHI 2003. *ACM Conf Hum Fact Comput Syst* 5(1):169–76.

Manning, H. 2002. Reflections: Must the sale end? *Interactions* 9(6):56. ACM.

Medlock, M., D. Wixon, M. McGee, and D. Welsh. 2005. The rapid iterative test and evaluation method: Better products in less time. In *Cost-Justifying Usability: An Update for the Information Age*, eds. R. Bias and D. Mayhew, 489–517. Morgan Kaufman.

Molich, R., M. R. Ede, K. Kaasgaard, and B. Karyukin. 2004. Comparative usability evaluation. *Behav Inf Technol* 23(1):65–74.

Molich, R., and J. Nielsen. 1990. Improving a human–computer dialogue. *Commun ACM* 33(3):338–48.

Muller, M. J., A. McClard, B. Bell, S. Dooley, L. Meiskey, J. A. Meskill, R. Sparks, and D. Tellam. 1995. Validating an extension to participatory heuristic evaluation: Quality of work and quality of life. In *Proc. ACM CHI'95 Conference on Human Factors in Computing Systems (Conference Companion)*, eds. I. Katz, R. Mack, and L. Marks, 115–6. New York: ACM Press.

Nielsen, J. 1992. Finding usability problems through heuristic evaluation. In *Proc. ACM CHI'92 Conf.*, eds. P. Bauersfeld, J. Bennett, and G. Lynch, 373–80. New York: ACM press.

Nielsen, J. 1993. *Usability Engineering*. San Francisco: Morgan Kaufmann.

Nielsen, J. 1994a. Enhancing the explanatory power of usability heuristics. In *Proc. CHI'94 Conference on Human Factors in Computing Systems*, eds. B. Adelson, S. Dumais, and J. Olson, 152–8. New York: ACM Press.

Nielsen, J. 1994b. Heuristic evaluation. In *Usability Inspection Methods*, eds. J. Nielsen and R. L. Mack, 25–62. New York: John Wiley & Sons.

Nielsen, J., and T. K. Landauer. 1993. A mathematical model of the finding of usability problems. In *Proc. INTERCHI'93 Conf. on Human Factors in Computing Systems*, eds. S. Ashlund, K. Mullet, A. Henderson, E. Hollnagel, and T. White, 206–13. New York: ACM Press.

Nielsen, J., and R. Molich. 1990. Heuristic evaluation of user interfaces. In *Proceedings of ACM CHI'90 Conference on Human Factors in Computing Systems*, eds. J. Carrasco and J. Whiteside, 249–56. New York: ACM Press.

Norman, D. A. 1986. Cognitive engineering. In *User Centered System Design: New Perspectives on Human–Computer Interaction*, eds. D. A. Norman and S. W. Draper, 31–61. Hillsdale, NJ: Lawrence Erlbaum Associates.

Nørgaard, M., and K. Hornbæk. 2006. What do usability evaluators do in practice? An explorative study of think-aloud testing. In *ACM Conference on Designing Interactive Systems*, 209–18. New York: ACM Press.

Polson, P. G., and C. H. Lewis. 1990. Theory-based design for easily learned interfaces. *Hum Comput Interact* 5(2–3):191–220.

Polson, P. G., C. Lewis, J. Rieman, and C. Wharton. 1992. Cognitive walkthroughs: A method for theory-based evaluation of user interfaces. *Int J Man Mach Stud* 36(5):741–73.

Rieman, J., S. Davies, D. C. Hair, M. Esemplare, P. Polson, and C. Lewis. 1991. An automated cognitive walkthrough. In *Proc. ACM CHI'91 Conf. on Human Factors in Computing Systems*, eds. S. P. Robertson, G. M. Olson, and J. S. Olson, 427–8. New York: ACM Press.

Rosenbaum, S. 2008. The future of usability evaluations: Increasing impact on value. In *Maturing Usability: Quality in Software, Interaction and Value*, eds. E. L. C. Law, E. Hvannberg, and G. Cockton, 344–78. London: Springer.

Rosenbaum, S., J. A. Rohn, and J. Humburg. 2000. A toolkit for strategic usability: Results from workshops, panels, and surveys. In *Proceedings of ACM CHI 2000 Conference on Human Factors in Computing Systems*, eds. R. Little and L. Nigay, 337–44. New York: ACM Press.

Rowley, D. E., and D. G. Rhoades. 1992. The cognitive jogthrough: A fast-paced user interface evaluation procedure. In *Proceedings of ACM CHI'92 Conference on Human Factors in Computing Systems*, eds. P. Bauersfeld, J. Bennett, and G. Lynch, 389–95. New York: ACM Press.

Scapin, D. L. 1990. Organizing human factors knowledge for the evaluation and design of interfaces. *Int J Hum Comput Interact* 2(3):203–29.

Scapin, D. L., and J. M. C. Bastien. 1997. Ergonomic criteria for evaluating the ergonomic quality of interactive systems. *Behav Inf Technol* 16(4/5):220–31.

Sears, A. 1997. Heuristic walkthroughs: Finding the problems without the noise. *Int J Hum Comput Interact* 9(3):213–34.

Sears, A., and D. Hess. 1999. Cognitive walkthroughs: Understanding the effect of task description detail on evaluator performance. *Int J Hum Comput Interact* 11(3):185–200.

Somervell, J., and D. S. McCrickard. 2005. Better discount evaluation: Illustrating how critical parameters support heuristic creation. *Interact Comput* 17(5):592–612.

Spencer, R. 2000. The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 353–9. New York: ACM.

Spool, J., and W. Shroeder. 2001. Testing web sites: Five users is nowhere near enough. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 285–6. New York: ACM.

Suchman, L. 1987. *Plans and Situated Actions: The Problem of Human-Machine Communication*. New York: Cambridge University Press.

Sy, D. 2007. Adapting usability investigations for agile user-centered design. *J Usability Stud* 2(3):112–32.

Theofanos, M., and W. Quesenbery. 2005. Towards the design of effective formative test reports. *J Usability Stud* 1(1):27–45.

Uldall-Espersen, T., E. Frøkjær, and K. Hornbæk. 2008. Tracing impact in a usability improvement process. *Interact Comput* 20(1):48–63.

Venturi, T., J. Troost, and T. Jokela. 2006. People, organizations, and processes: An inquiry into the adoption of user-centered design in industry. *International Journal of Human-Computer Interaction*, 21(2):219–238.

Wharton, C., J. Rieman, C. Lewis, and P. Polson. 1994. The cognitive walkthrough: A practitioner's guide. In *Usability Inspection Methods*, eds. J. Nielsen and R. L. Mack, 105–40. New York: John Wiley & Sons.

Wixon, D. 2003. Evaluating usability methods: Why the current literature fails the practitioner. *Interactions* 10(4):28–34.

Woolrych, A. 2001. *Assessing the Scope and Accuracy of the Usability Inspection Method Heuristic Evaluation*, MPhil, University of -Sunderland. http://osiris.sunderland.ac.uk/,cs0awo/downloadable%20documents.htm (accessed 21/12/05).

Woolrych, A., and G. Cockton. 2000. Assessing heuristic evaluation: Mind the quality, not just percentages. In *Proceedings of British HCI Group HCI 2000 Conference*, Vol. 2, eds. S. Turner and P. Turner, 35–6. London: British Computer Society.

Woolrych, A., and G. Cockton. 2002. Testing a conjecture based on the DR-AR model of usability inspection method effectiveness. In *People and Computers XVI: Memorable yet Invisible*, Vol. 2, eds. H. Sharp, P. Chalk, J. LePeuple, and J. Rosbottom, 30–3. London: British Computer Society.

Woolrych, A., G. Cockton, and M. Hindmarch. 2004. Falsification testing for usability inspection method assessment. In *Proceedings of HCI 2004*, Vol. 2, eds. A. Dearden, and L. Watts, 137–40. Bristol, UK: Research Press International.

Woolrych, A., G. Cockton, and M. Hindmarch. 2005. Knowledge resources in usability inspection. In *Proceedings of HCI 2005*, Vol. 2, eds. L. Mackinnon, O. Bertelsen and N. Bryan-Kinns, 15–20.

Woolrych, A., K. Hornbæk, E. Frøkjær, and G. Cockton. 2012. Ingredients and meals rather than recipes: A proposal for research that does not treat usability evaluation methods as indivisible wholes. *International Journal of Human-Computer Interaction* 27(10):940–70.

Young, R. M., and T. Simon. 1987. Planning in the context of human–computer interaction. In *People and Computers III*, eds. D. Diaper and R. Winder, 363–70. Cambridge, UK: Cambridge University Press.

57 Model-Based Evaluation

David Kieras

CONTENTS

57.1 INTRODUCTION

57.1.1 What Is Model-Based Evaluation?

Model-based evaluation is using a *model* of how a human would use a proposed system to obtain predicted usability measures by calculation or simulation. These predictions can replace or supplement empirical measurements obtained by user testing. In addition, the content of the model itself conveys useful information about the relationship between the user's task and the system design.

57.1.2 Organization of This Chapter

This chapter will first argue that model-based evaluation is a valuable supplement to conventional usability evaluation and then survey the current approaches for performing model-based evaluation. Because of the considerable technical detail involved in applying model-based evaluation techniques, this chapter cannot include "how to" guides on the specific modeling methods, but they are all well documented elsewhere. Instead, this chapter will present several high-level issues in constructing and using models for interface evaluation and comment on the current approaches in the context of those issues. This will assist the reader in deciding whether to apply a model-based technique, which one to use, what problems to avoid, and what benefits to expect. Somewhat more detail will be presented about one form of model-based evaluation, Goals, Operators, Methods, and Selection rules (GOMS) models, which is a well developed, relatively simple, and "ready to use" methodology applicable to many interface design problems. A set of concluding recommendations will summarize the practical advice.

57.1.3 Why Use Model-Based Evaluation?

Model-based evaluation can be best viewed as an alternative way to implement an iterative process for developing a usable system. This section will summarize the standard usability process and contrast it with a process using model-based evaluation.

57.1.3.1 Standard Usability Design Process

In simplified and idealized form, the standard process for developing a usable system centers on user testing of prototypes that seeks to compare user performance to a specification or identify problems that impair learning or performance. After performing a task analysis and choosing a set of benchmark tasks, an interface design is specified based on intuition and guidelines both for the platform/application style and usability. A prototype of some sort is implemented and then a sample of representative users attempts to complete the benchmark tasks with the prototype. Usability problems are noted, such as excessive task completion time or errors, being unable to complete a task, or confusion over what to do next. If the problems are serious enough, the prototype is revised and a new user test is conducted. At some point, the process is terminated and the product is completed, either because no more serious problems have been detected or because there is not enough time or money for further development. See Dumas (this volume) for a complete presentation of this iterative user-testing methodology.

The standard process is a straightforward, well-documented methodology with a proven record of success (Landauer 1995). The guidelines for user interface design, together with knowledge possessed by those experienced in interface design and user testing, add up to a substantial accumulation of wisdom on developing usable systems. There is no doubt that if this process were applied more widely and thoroughly, the result would be a tremendous improvement in software quality. User testing has always been considered the "gold standard" for usability assessment. However, it has some serious limitations—some practical and others theoretical.

57.1.3.2 Practical Limitations of User Testing

A major practical problem is that user testing can be too slow and expensive to be compatible with current software development schedules, and so a focus of human–computer interaction (HCI) research for many years has been a way to tighten the iterative design loop. For example, better prototyping tools allow prototypes to be developed and modified more rapidly. Clever use of paper mockups or other early user input techniques allows important issues to be addressed before making the substantial investment in programming a prototype. The so-called inspection evaluation methods seek to replace user testing with other forms of evaluation, such as expert surveys of the design or techniques such as cognitive walkthroughs (Cockton et al. this volume).

If user testing is really the best method for usability assessment, then it is necessary to come to terms with the unavoidable time and cost demands of collecting behavioral data and analyzing it, even in the rather informal manner that normally suffices for user testing. For example, if the system design were substantially altered on iteration, it would be necessary to retest the design with a new set of test users. While it is hoped that the testing process finds fewer important problems with each iteration, the process does not get any faster with each iteration—the same adequate number of test users must perform the same adequate number of representative tasks and their performance assessed.

The cost of user testing is especially pronounced in expert-use domains, where the user is somebody like a physician, a petroleum geologist, or an engineer. Such users are few and their time is valuable. This may make relying on user testing too costly to adequately refine an interface. A related problem is evaluating software that is intended to serve experienced users especially well. Assessing the quality of the interface requires a very complete prototype that can be used in a realistic way for an extended period of time so that the test users can become experienced. This drives up the cost of each iteration, because the new version of the highly functional prototype must be developed and the lengthy training process has to be repeated. Other design goals can also make user testing problematic: consider developing a pair of products for which skill is supposed to transfer from one to the other. Assessing such transfer requires prototyping both products fully enough to train users on the first and then training them on the second, to see if the savings in training time are adequate. Any design change in either of the products might affect the transfer, and thus require a repeat test of the two systems. This double-dose of development and testing effort is probably impractical except in critical domains, where the additional problem of testing with expert users will probably appear.

57.1.3.3 Theoretical Limitations of User Testing

From the perspective of scientific psychology, the user testing approach takes very little advantage of what is known about human psychology, and thus lacks grounding in psychological theory. Although scientific psychology has been underway since the late 1800s, the only concepts relied on by user testing are a few basic concepts of how to collect behavioral data. Surely more is known about human psychology than this! The fact is that user testing methodology would work even if there was no systematic scientific knowledge of human psychology at all—as long as the designer's intuition leads in a reasonable direction on each iteration, it suffices merely to revise and retest until no more problems are found. While this is undoubtedly an advantage, it does suggest that user testing may be a relatively inefficient way to develop a good interface.

This lack of grounding in psychological principles is related to the most profound limitation of user testing: it lacks a systematic and explicit representation of the knowledge developed during the design experience; such a representation could allow design knowledge to be accumulated, documented, and systematically reused. After a successful user testing process, there is no representation of how the design "works" psychologically to ensure usability—there is only the final design itself, as described in specifications or in the implementation code. These descriptions normally have no theoretical relationship to the user's task or the psychological characteristics of the user. Any change to the design, or to the user's tasks, might produce a new and different usability situation, but there is no way to tell what aspects of the design are still relevant or valid. The information on why the design is good, or how it works for users, resides only in the intuitions of the designers. While designers often have outstanding intuitions, we know from the history of creations such as the medieval cathedrals that intuitive design is capable of producing magnificent results, but is also routinely guilty of costly over-engineering or disastrous failures. We now know that medieval structures benefitted from the happy accident that building in stone is relatively fool-proof; intuitive design proved dangerous with more modern materials (Gordon 1978). Perhaps, because of the complex and rapidly developing nature of computer software, the experience of the usability field seems to be that the ordinary software developer's intuition is quite unreliable, and so the only widely accepted recipe for avoiding design failure is slow and expensive iterative design based on user testing.

57.1.3.4 Model-Based Approach

The goal of model-based evaluation is to get some usability results before implementing a prototype or testing with human subjects. The approach uses a *model* of the HCI situation to represent the interface design and produce predicted measurements of the usability of the interface. Such models are also termed *engineering models* or *analytic models* for usability. The model is based on a detailed description of the proposed design and a detailed task analysis; it explains how the users will accomplish the tasks by interacting with the proposed interface and uses psychological theory and parametric data to generate the predicted usability metrics. Once the model is built, the usability predictions can be quickly and easily obtained by calculation or by running a simulation. Moreover, the implications of variations on the design can be quickly explored by making the corresponding changes in the model. Since most variations are relatively small, a circuit around the revise/evaluate iterative design loop is typically quite fast once the initial model-building investment is made. Thus, unlike user testing, iterations generally get faster and easier as the design is refined.

In addition, the model itself summarizes the design and can be inspected for insight into how the design supports (or fails to support) the user in performing the tasks. Depending on the type of model, components of it may be reusable not in just different versions of the system under development, but in other systems as well. Such a reusable model component captures a stable feature of human performance, task structures, or interaction techniques; characterizing them contributes to our scientific understanding of HCI.

The basic scheme for using model-based evaluation in the overall design process is that iterative design is done first using the model and then by user testing. In this way, many design decisions can be worked out before investing in prototype construction or user testing. The final user testing process is required for two reasons. First, the available modeling methods only cover certain aspects of usability; at this time, they are limited to predicting the sequence of actions, the time required executing the task, and certain aspects of the time required to learn how to use the system. Thus, user testing is required to cover the remaining aspects. Second, since the modeling process is necessarily imperfect, user testing is required to ensure that some critical issue has not been overlooked. If the user testing reveals major problems along the lines of a fundamental error in the basic concept of the interface, it will be necessary to go back and reconsider the entire design; again model-based iterations can help address some of the issues quickly. Thus, the purpose of the model-based evaluation is to perform some of the design iterations in a lower-cost, higher-speed mode before the relatively slow and expensive user testing.

57.1.3.5 What "Interface Engineering" Should Be

Model-based evaluation is not the dominant approach to user interface development; most practitioners and academics seem to favor some combination of user testing and inspection methods. Some have tagged this majority approach as a form of "engineering." However, even a cursory comparison to established engineering disciplines (e.g., in an engineering handbook such as Merritt, Loftin, and Ricketts 1996) makes it clear that conventional approaches to user interface design and evaluation has little resemblance to an engineering discipline. In fact, model-based evaluation is a deliberate attempt to develop and apply true engineering methods for user interface design. The following somewhat extended analogy will help clarify the distinction and explain the need for further research in modeling techniques.

If civil engineering were done with iterative empirical testing, bridges would be built by erecting a bridge according to an intuitively appealing design and then driving heavy trucks over it to see if it cracks or collapses. If it does, it would be rebuilt in a new version (e.g., with thicker girders) and the trial repeated; the iterative process continues with additional guesses until a satisfactory result is obtained. Over time, experienced bridge-builders would develop an intuitive feel for good designs and how strong the structural members need to be, and so will often guess right. However, time and cost pressures will probably lead to cutting the process short by favoring conservative designs that are likely to work, even though they might be unnecessarily clumsy and costly.

Although very early bridge-building undoubtedly proceeded in this fashion, modern civil engineers do not build bridges by iterative testing of trial structures. Rather, in the early 1800s under the combined stimulus of developing mathematics, new materials, more challenging applications, and design failures (Beckett 1984; Buchanan 1989; Gordon 1978; Petrosky 1985; Pugsley 1976), engineering researchers and practitioners began to develop a body of scientific theory on the behaviors of structures and forces and a body of principles and parametric data on the strengths and limitations of bridge-building materials. By the mid-1800s, important bridges were designed based on mathematical models with extensive calculations; the models for new design concepts were sometimes verified using physical models of sub-components of the overall structure. From this theory and data, civil engineers can quickly construct models in the form of equations or computer simulations that allow them to evaluate the quality of a proposed design without having to physically construct a bridge (e.g., see Merrit, Loftin, and Ricketts 1996). Modeling is a central part of the design process.

The bridge is not built until the design has been tested and evaluated based on the models, and the models themselves checked for correctness, and the new bridge almost always performs correctly; enough so that final testing of the completed structure is not routinely done. When final testing has been done, it has served as final verification of a successful design, not as an iteration along the way to a final design. It is fair to say that starting in the mid-1800s bridges were routinely and successfully built in a single physical iteration!

Thus, an investment in theory development and measurement enables engineers to replace an empirical iterative process with a theoretical iterative process that is much faster and cheaper per iteration. Of course, the modeling process is fallible, and so designers include conservative safety factors in the analysis to cover errors or defects in materials and maintenance and occasionally make mistakes that result in incorrect designs. More importantly, occasionally the model for a new design is found to be seriously inaccurate—such as missing an important factor relevant to an innovative design—and a spectacular and deadly design failure is the result. While the failures are prominent, relative to the number of bridges built, design failures have been rare in the last century.

The claim is not that using engineering models is perfect or infallible, only that it saves time and money and thus allows designs to be more highly refined. In short, more design iterations results in better designs, and more iterations are possible if some of them can be done very cheaply using models.

Moreover, the theory and the model summarize the design and explain why the design works well or poorly. The theoretical analysis identifies the weak and strong points of the design, giving guidance to the designer where intuition can be applied to improve the design; a new analysis can then test whether the design has actually been improved. Engineering analysis does not result in simply static repetition of proven ideas. Rather, it enables more creativity because it is now possible to cheaply and quickly determine whether a new concept will work. Thus, novel and creative concepts for bridge structures have steadily appeared once the engineering models were developed—as was seen even in the nineteenth century and continue today.

Correspondingly, model-based evaluation of user interfaces is simply the rigorous and science-based techniques for how to evaluate user interfaces without user testing; it likewise relies on a body of theory and parametric data to generate predictions of the performance of an engineered artifact and explain why the artifact behaves as it does. While true interface engineering is nowhere as advanced as bridge engineering, useful techniques have been available for some time and should be more widely used. As model-based evaluation becomes more developed, it will become possible to rely on true engineering methods to handle most of the routine problems in user interface design, with considerable savings in cost and time and with reliably higher quality. As has happened in other branches of engineering, the availability of powerful analysis tools means that the designer's energy and creativity can be unleashed to explore fundamentally new applications and design concepts.

The fact that these methods are not as widely used as they could be should not be taken as evidence against their utility. Even in bridge engineering, the mathematical analyses were considered very difficult when first introduced, and many prominent engineers deprecated them in favor of more intuitive "practical" approaches (Gordon 1978; Buchanan 1989). Only as the design problems became more difficult did the value of mathematical methods become compelling, and after several decades, they became standard in engineering training and practice. Thus, we can expect that true user interface engineering methodology will take some time to be adopted, even as research continues.

57.1.4 Three Current Approaches

Research in HCI and allied fields has resulted in many models of HCI at many levels of analysis. This chapter restricts attention to approaches that have developed to the point that they have some claim, either practical or scientific, to being suitable for actual application in design problems. This section identifies three current approaches to modeling human performance that are the most relevant to model-based evaluation for system and interface design. These are task network models, cognitive architecture models, and GOMS models.

57.1.4.1 Task Network Models

In task network models, task performance is modeled in terms of a PERT-chart-like network of processes. Each process starts when its prerequisite processes have been completed and has an assumed distribution of completion times. This basic model can be augmented with arbitrary computations to determine the completion time and what its symbolic or numeric inputs and outputs should be. Note that the processes are usually termed “tasks,” but they need not be human-performed at all, but can be machine processes instead. In addition, other information, such as workload or resource parameters can be attached to each process. Performance predictions are obtained by running a Monte-Carlo simulation of the model activity, in which the triggering input events are generated either by random variables or by task scenarios. A variety of statistical results, including aggregations of workload or resource usage, values can be readily produced. The classic SAINT (Chubb 1981) and the commercial MicroSaint tool (Laughery 1989) are prime examples. These systems originated in applied human factors and systems engineering and are heavily used in system design, especially for military systems.

57.1.4.2 Cognitive Architecture Models

Cognitive architecture systems are surveyed by Byrne (this volume). These systems consist of a set of hypothetical interacting perceptual, cognitive, and motor components assumed to be present in the human and whose properties are based on empirical and theoretical results from scientific research in psychology and allied fields. The functioning of the components and their interactions are typically simulated with a computer program, which in effect produces a *simulated human* performing in a *simulated task environment* that supplies inputs (stimuli) to the simulated human, and reacts to the outputs (responses) produced by the simulated human. Tasks are modeled primarily by programming the cognitive component according to a task analysis, and then performance predictions are obtained by running the simulation using selected scenarios to generate the input events in the task. Because these systems are serious attempts to represent a theory of human psychological functions, they tend to be rather complex and are primarily used in basic research projects; there has been very limited experience in using them in actual design settings.

57.1.4.3 GOMS Models

GOMS models are the original approach to model-based evaluation in the computer user interface field; both the model-based evaluation approach and GOMS models were presented as methods for user interface design in the seminal Card, Moran, and Newell (1983) presentation of the psychology of HCI. They based the GOMS concept on the theory of human problem-solving and skill acquisition. In brief, GOMS models describe the knowledge of procedures that a user must have to operate a system, The acronym and the approach can be summarized as follows: the user can accomplish certain goals (G) with the system; operators (O) are the basic actions that can be performed on the system such as striking a key or finding an icon on the screen; methods (M) are sequences of operators that when executed accomplish a goal; selection rules (S) describe which method should be used in which situation to accomplish a goal, if there is more than one available. Constructing a GOMS model involves writing out the methods for accomplishing the task goals of interest and then calculating predicted usability metrics from the method representation.

There are different forms of GOMS models, systematized by John and Kieras (1996a,b), which represent the methods at different levels of detail and whose calculations can range in complexity from simple hand calculations to full-fledged simulations. John and Kieras pointed out that the different forms can be viewed as being based on different simplified cognitive architectures that make the models easy to apply to typical interface design problems and insulate the model-builder from many difficult theoretical issues. More so than any other model-based approach, GOMS models have a long and well-established track record of success in user interface design, although they are not used as widely as their simplicity and record would justify. Although still under development by researchers, GOMS models are emphasized in this chapter because in some forms they are a “ready to use” modeling methodology. Section 57.4 will describe their rationale more completely, but the reader is referred to John and Kieras (1996a,b) for a thorough discussion.

57.2 THEORETICAL BASIS FOR CHOOSING A MODEL-BASED EVALUATION TECHNIQUE

This section presents several key issues concerning the theoretical foundations of model-based evaluation, concerning the basic sources of information and applicability of the modeling approach. When choosing or evaluating a technique for model-based evaluation, the potential user should consider these issues; the techniques differ widely in how well they handle certain fundamental questions. The next section will focus on the practical problems of applying a modeling technique once it has been chosen. In both sections, the three basic approaches to model-based evaluation are commented on as appropriate. Advice is given to both the user of model-based evaluation and the developer of model-based techniques.

57.2.1 Psychological Constraints Are Essential

The concept of model-based evaluation in system design has a long history and many different proposed methods (for early surveys, see Pew et al. 1977; McMillan et al. 1989; Elkind et al. 1989). However, the necessary scientific basis for genuinely powerful models has been slow to develop. The key requirement for model-based evaluation is that building a model to evaluate a design must be a routine, production, or engineering activity and not a piece of basic scientific research on how human psychological factors are involved in a particular computer usage situation. This means that the relevant psychological science must not only be developed first, but also then systematized and encapsulated in the modeling methodology itself. That is, a modeling methodology must provide *constraints* on the content and form of the model, and these constraints must provide the psychological validity of the model as a predictor of human performance. In other words, if the model builder can do essentially anything in the modeling system, then the only way the resulting model can be psychologically valid is if the model builder does all the work to construct a valid psychological theory of human cognition and behavior in the task and then ensure that the constructed model accurately reflects this theory.

Of course, it takes tremendous time, effort, and training to construct original psychological theory, far more than should be necessary for most interface design situations. Although the decisions in truly novel or critical design situations might require some fundamental psychological research, most interface design situations are rather routine: the problem is to match a computer system to the user's tasks using known interface design concepts and techniques. It should not be necessary to be an expert researcher in human cognition and performance to carry this out.

Thus, the key role of a modeling system is to provide constraints based on the psychological science, so that a model constructed within the system has a useful degree of predictive validity. In essence, simply by using the modeling system according to its rules, the designer must be able to construct a scientifically plausible and usefully accurate model "automatically."

A simple series of examples will help make the point: computer user interfaces involve typing of arbitrary strings of text on the keyboard and pointing with a mouse. The time required to type on the keyboard and to point with a mouse is fairly well documented. If task execution times are of interest, an acceptable modeling system should include these human performance parameters so that the interface designer does not have to collect them or guess them.

Furthermore, because both hands are involved in typing strings of text, users cannot type at the same time as they move the mouse cursor; these operations must be performed sequentially, taking rather more time than if they could be done simultaneously. A modeling system should make it impossible to construct a model of an interface that overzealously optimizes execution speed by assuming that the user could type strings and point simultaneously; the sequential constraint should be enforced automatically. A high-quality modeling system would not only enforce this constraint, but also automatically include the time costs of switching between typing and pointing, such as the time to move the hand between the mouse and the keyboard. There are many such constraints on human performance, some of them quite obvious, as in these examples, and some very subtle. A good modeling system will represent these constraints in such a way that they are automatically taken into account in how the model can be constructed and used. Because of the subtleties involved, computational tools are especially valuable for constructing and using models because they can help enforce the psychological constraints and make it easier for the model-builder to work within them.

57.2.2 Brief History of Constraints in Modern Psychological Theory

Theoretical constraints are not easy to represent or incorporate; a coherent and rigorous theoretical foundation is required to serve as the substrate for the network of constraints, and suitable foundations were not constructed until fairly recently. Through most of second half of the twentieth century, psychological theory was mired in a rather crude form of information-processing theory, in which human activity was divided into information-processing stages, such as perception, memory, decision making, and action, usually depicted as a flowchart with a box for each stage, various connections between the boxes, and perhaps with some fairly simple equations that described the time required for each stage or the accuracy of its processing. However, there was little constraint on the possible data contained in each box or the operations performed there; a box could be of arbitrary complexity, and no actual explicit mechanism had to be provided for any of them. Such models were little more than a "visual aid" for theories posed in the dominant forms of informal verbal statements or rather abstract mathematical equations. Later, many researchers began to construct computer simulations of these "box models," which provided more flexibility than traditional mathematical models and also contributed more explicitness and rigor than traditional verbal models. But still the operations performed in each box were generally unstructured and arbitrary.

An early effort at model-based evaluation in this theoretical mode appears in the famous human operator simulator (HOS) system (see Wherry 1976; Pew et al. 1977; Strieb and Wherry 1979; Lane et al. 1981; Glenn, Zaklad, and Wherry 1982; and Harris, Iavecchia, and Bittner 1988; Harris, Iavecchia, and Dick 1989). HOS contained a set of *micromodels* for low-level perceptual, cognitive, and motor activities, invoked by task-specific programs written in a special-purpose procedural programming language called HOPROC (human operator procedures language). The micromodels included such things as Hick's and Fitts' Law, formulas for visual recognition time, a model of short-term memory retention, and formulas for calculating the time required for various motor actions such as pushing buttons and walking. The effort was ambitious and the results impressive, but in a real sense, HOS was ahead of its time. The problem was that psychological theory was not well enough developed at the time to provide a sound foundation for such a tool; the developers were basically trying to invent a cognitive architecture good enough for practical application before the concept had been developed in the scientific community. Interestingly, the spirit of the HOPROC language lives on in the independently developed notations for some forms of GOMS models. In addition, the scientific base for the micromodels was in fact very sparse at the time, and many of them are currently out of date empirically and theoretically. HOS appears to have been subsumed into some commercial modeling systems; for example, a task network version of HOS is available from Micro Analysis and Design, Inc. (http://www.maad.com/), and its micromodels are used in their integrated performance modeling environment (IPME), as well as CHI System's COGNET/IGEN (http://www.chiinc.com/).

The task network models also originated in this box-model mode of psychology theory and show it in their lack of psychological constraints; their very generality means they contribute little built-in psychological validity. Even if the HOS micromodels are used, the flexibility of the modeling system means that model-builders themselves must identify the psychological processes and constraints involved in the task being modeled and program them into the model explicitly.

Led by Anderson (1983) and Newell (1990), researchers in human cognition and performance began to construct models using a *cognitive architecture* (see Byrne, this volume). Cognitive architecture parallels the concept of computer architecture: a cognitive architecture specifies a set of fixed mechanisms, the "hardware," that comprise the human mind. To construct a model for a specific task, the researcher "programs" the architecture by specifying a psychological strategy for doing the task, the "software" (specifying parameter value settings and information in memory might be involved as well.) The architecture provides the coherent theoretical framework within which the processes and constraints can be proposed and given an explicit and rigorous definition. Several proposed cognitive architectures exist in the form of computer simulation packages in which programming the architecture is done in the form of production systems, collections of modular if-then rules, which have proved to be an especially good theoretical model of human procedural knowledge. Developing these architectures, and demonstrating their utility, is a continuing research activity (see Byrne, this volume). Not surprisingly, they all have a long way to go before they accurately incorporate even a subset of the human abilities and limitations that appear in an HCI design context.

The psychological validity of a model constructed with a cognitive architecture depends on the validity of both the architecture and the task-specific programming, and so it can be difficult to assign credit or blame for success or failure in modeling an individual task. However, the fixed architecture and its associated parameters are supposed to be based on fundamental psychological mechanisms that are required to be invariant across all tasks, while the task-specific programming is free to vary with a particular modeled task. To the extent that the architecture is correct, one should be able to model any task simply by programming the architecture using only task-analytic information and supplying a few task-specific parameters. The value of such architectures lies in this clear division between universal and task-specific features of human cognition; the model builder should be free to focus solely on the specific task and system under design and let the architecture handle the psychology.

A key property of most current architectures is the insistence on a small number of fundamental mechanisms that provide a comprehensive and coherent system. For example, several of the scientifically successful cognitive architectures require that all cognitive processing must be expressed in the form of production rules that can include only certain things in their conditions and actions. These rules control all the other components in the architecture, which in turn have strictly defined and highly limited capabilities. These highly constrained systems have been successful in a wide range of modeling problems, showing that to be useful in both scientific and practical prediction, the possible models must be constrained—too many possibilities are not helpful, but harmful.

Achieving this goal in psychological research is a daunting challenge. What about the practical sphere? In fact, the role of architectural constraints in some of the extant commercial modeling systems is problematic. The task network models basically have such an abstract representation that there is no straightforward way for architectural assumptions to constrain the modeling system. Once one has opted for representing human activity as a set of arbitrary interconnected task processes, there is no easy way to somehow impose more constrained structure and mechanism on the system. Attempting to do so simply creates more complexity in the modeling problem—the modeler must figure out how to *underuse* the *over-general* capabilities of the system in just the right way.

At the other extreme, a modeling system that attempts to cover the higher aspects of human cognition by providing complex architectures, such as COGNET/IGEN (see Zachary et al. 2000, for a relatively complete description) can have the form of a cognitive architecture, but as argued more below, as a practical matter, it is difficult to analyze these aspects in a routine interface design problem. From the point of view of cognitive architectures and the constraints supplied by the architecture, the modeling approaches described in this chapter, as currently implemented, span the range from little or no architectural content or constraints (the task network systems) to considerable architectural complexity and constraints (the cognitive-architecture systems). GOMS models occupy an intermediate position: they assume a simplified, but definitely constraining, cognitive architecture that allows them to be applied easily by interface designers and still produce usefully accurate results. But at the same time, they are less flexible than the modeling systems at the other extremes.

57.2.3 Modeling Cognitive versus Perceptual-Motor Aspects of a Design

As pointed out by Byrne (this volume), cognitive architectures have lately begun to incorporate not just proposed cognitive mechanisms, but also proposals for perceptual and motor mechanisms that act as additional sources of constraint on performance. Calling these a "cognitive" architecture is something of a misnomer, since perceptual and motor mechanisms are normally distinguished from cognitive ones. However, including perceptual and motor constraints is actually a critical requirement for modeling user interfaces; this follows from the traditional characterization of HCI in terms of the interactive cycle (Norman 1986). The user sees something on the screen if they are looking in the right place and can sense and recognize it, involving the perceptual system and associated motor processes such as eye movements. The user decides what to do, an exclusively cognitive activity, and then carries out the decision by performing motor actions that are determined by the physical interaction devices that are present and may also involve the perceptual system, such as visual guidance for mouse pointing.

Occasionally, the cognitive processes of deciding what to do next can dominate the perceptual and motor activities. For example, one mouse click might bring up a screen containing a single number, such as a stock price, and the user might think about it for many minutes before simply clicking on a "buy" or "sell" button. But in many user interface settings, users engage in a stream of routine activities that require only relatively simple cognitive processing, and so the perceptual and motor actions take up most of the time and determine most of the task structure. Two implications follow from this thumbnail analysis.

57.2.3.1 Modeling Purely Cognitive Tasks Is Generally Impractical

Trying to model purely cognitive tasks such as human problem-solving, reasoning, or decision-making processes is extremely difficult because they are so open-ended and unconstrained (see also Landauer 1995). For example, there are a myriad possible ways in which people could decide to buy or sell a stock, and the nature of the task does not set any substantial or observable constraints on how people might make such decisions—stock decisions are based on everything from gut feel to transient financial situations, to detailed long-term analysis of market trends, and to individual corporate strategies. Trying to identify the strategy that a user population will follow in such tasks is not a routine interface design problem, but a scientific research problem, or at least a very difficult task analysis problem. Fortunately, a routine task analysis may produce enough information to allow the designer to *finesse* the problem, that is, side-step it or avoid having to confront it. For example, if one could determine what information the stock-trader needs to make the decisions and then make that information available in an effective and usable manner, the result will be a highly useful and usable system without having to understand exactly how users make their decisions.

57.2.3.2 Modeling Perceptual-Motor Activities Is Critical

A good modeling approach at a minimum must explicitly represent the perceptual and motor operations involved in a task. For most systems, the perceptual and motor activities involved in interacting with a computer take relatively well-defined amounts of time, are heavily determined by the system design, and frequently dominate the user's activity; leaving them out of the picture means that the resulting model is likely to be seriously inaccurate. For example, if two interface designs differ in how many visual searches or mouse points they logically require to complete a task, the one requiring fewer is almost certainly going to be faster to execute and will probably have a simpler task structure as well, meaning it will probably be easier to learn and less error-prone. Since perceptual-motor activities are relatively easy to model, it can be easy to get fairly reliable and robust model-based evaluation information in many cases. This means that any modeling approach that represents the basic timing and the structure of perceptual and motor activity entailed by an interface is likely to provide a good approximation to the basic usability characteristics of the interface. One reason why GOMS models work so well is that they allow the modeler to easily represent perceptual-motor activity.

57.2.4 Science Base Must Be Visible

Even though the modeling methodology encapsulates the constraints provided by psychological theory, it is critical that the psychological assumptions be accessible, justified, and intelligible. An architecture is the best way to do this, because the psychological assumptions are either hard-wired into the modeling system architecture or are explicitly stated in the task-specific programming supplied by the modeler. The basis for the task-specific programming is the task analysis obtained during the overall design process, and the basis for the architecture is a documented synthesis of the scientific literature.

The importance of the documented synthesis of the scientific literature cannot be overstated. The science of human cognition and performance that is relevant to system design is not at all "finished"; important new results are constantly appearing, and many long-documented phenomena are incompletely understood (e.g., Kieras 2009). Thus, any modeling system will have to be updated repeatedly as these theoretical and empirical issues are thrashed out, and it will have to be kept clear which results it incorporates and which it does not.

The commercial modeling tools have seriously lagged behind the scientific literature; while some conservatism would be desirable to damp out some of the volatility in scientific work, the problem is not just conservatism, but rather obsolescence, as in the case of the micromodels inherited from HOS. Perhaps these systems would still be adequate for practical work but, unfortunately, it is very difficult to get a scientific perspective on their adequacy because they have been neither described nor tested in forums and under ground rules similar to those used for mainstream scientific work in human cognition and performance. Thus, they have not been subject to the full presentation, strict review, criticism, and evolution that are characteristic of the cognitive architecture and GOMS model work. The practitioner should, therefore, greet the claims of commercial modeling system with healthy skepticism and developers of modeling systems should participate more completely in the open scientific process.

57.2.5 Value of Generativity

It is useful if a modeling method is *generative*, meaning that a single model can *generate* predicted human behavior for a whole class of scenarios, where a scenario is defined solely in terms of the sequence of input events or the specifications for a task situation, neither of which specifies the behavior the user is expected to produce. Many familiar modeling methods, including the Keystroke-Level type of GOMS model, are nongenerative, in that they start with a specific scenario in which the model builder has specified, usually manually, what the user's actions are supposed to be for the specified inputs. A nongenerative model predicts metrics defined only over this particular input–output sequence. To see what the results would be for a different scenario, a whole new model must be constructed (though parts might be duplicated). Since nongenerative modeling methods are typically labor intensive, involving a manual assignment of user actions to each input–output event, they tend to sharply limit how many scenarios are considered, which can be very risky in complex or critical design problems. Modern computer-based analytic tools can help overcome this limitation by making it easy to reuse parts of scenario, as in CogTool, described later.

An example of a sophisticated nongenerative modeling method is the CPM-GOMS models developed by Gray, John, and Atwood (1993) to model telephone operator tasks. These models decomposed each task scenario into a set of operations performed by perceptual, cognitive, and motor processors likes those proposed in the Card, Moran, and Newell (1983) model human processor. The sequential dependencies and time durations of these operations were represented with a PERT chart, which then specified the total task time and whose critical path revealed the processing bottlenecks in task performance. Such models are nongenerative in that a different scenario with a different pattern of events requires a different PERT chart to represent the different set of process dependencies. Since there is a chart for each scenario, predicting the time for a new scenario, or different interface design, requires creating a new chart to fit the new sequence of events. However, a new chart can often be assembled from templates or portions of previous charts, saving considerable effort (see John and Kieras 1996a,b for more detail).

However, if a model is generative, a single model can produce predicted usability results for any relevant scenario, just like a computer program for calculating the mean of a set of numbers can be applied to any specific set of values. A typical hierarchical task analysis (HTA) (see Annett et al. 1971; Kirwan and Ainsworth 1992) results in a generative representation, in that the HTA chart can be followed to perform the task in any subsumed situation. The forms of GOMS models that explicitly represent methods (see John and Kieras 1996a,b) are also generative. The typical cognitive-architecture model is generative in that it is programmed to perform the cognitive processes necessary to decide how to respond appropriately to any possible input that might occur in the task. In essence, the model programming expresses the general procedural knowledge required to perform the task, and the architecture, when executing this procedural knowledge, supplies all the details; the result is that the model responds with a different specific time sequence of actions to different specific situations.

For example, Kieras, Wood, and Meyer (1997) used a cognitive architecture to construct a production-rule model of some of the telephone operator tasks studied by Gray, John, and Atwood (1993). Because the model consisted of a general "program" for doing the tasks, it would behave differently depending on the details of the input events; for example, greeting a customer differently depending on information on the display and punching function keys and entering data depending on what the customer says and requires. Thus, the specific behavior and its time course of the model depend on the specific inputs, in a way expressed by a single set of general procedures.

A generative model is typically more difficult to construct initially, but because it is not bound to a specific scenario, it can be directly applied to a large selection of scenarios to provide a comprehensive analysis of complex tasks. The technique is especially powerful if the model runs as a computer simulation in which there is a *simulated device* that represents how the scenario data results in specific display events and governs how the system will respond to the user and the *simulated human*, which is the model of how the user will perform the task. The different scenarios are just the input data for the simulation, which produces the predicted behavior for each one. Furthermore, because generative models represent the procedural knowledge of the user explicitly, they readily satisfy the desirable property of models described earlier: the content of a generative model can be inspected to see how a design "works" and what procedures the user must know and execute.

57.2.6 Role of Detail

In the initial presentation above, the reader may have noticed the emphasis on the role of detailed description, both of the user's task and the proposed interface design. Modeling has sometimes been criticized because it appears to be unduly labor intensive. Building a model and using it to obtain predictions may indeed involve substantial detail work. However, working out the details about the user's task and the interface design is, or should be, a necessary part of any interface design approach; the usability lies in the details, not the generalities (Whiteside et al. 1985). If the user's task has not been described in detail, chances are that the task analysis is inadequate and a successful interface will be more difficult to achieve; extra design iterations may be required to discover and correct deficiencies in the original understanding of the user's needs. If the interface designer has not worked out the interface design in detail, the prospects of success are especially poor. The final form of an interface reflects a mass of detailed design decisions; these should have been explicitly made by an interface designer whose focus is on the user, rather than the programmers who happen to write the interface code. So the designer has to develop this detail as part of any successful design effort. In short, using model-based evaluation does not require any more detail than should be available anyway; it just requires that this detail be developed more explicitly and earlier than is often the case. Tools such as CogTool, described below, are promising because they evnable UI designers to easily explore alternative design prototypes in detail and also calculate usability metrics based on these details.

57.2.6.1 Cognitive Architectures Are Committed to Detail

Cognitive architecture systems are primarily research systems dedicated to synthesizing and testing basic psychological theory. Because they have a heavy commitment to characterizing the human cognitive architecture in detail, they naturally work at an extremely detailed level. The current cognitive architecture systems differ widely in the extent to which they incorporate the most potent source of practical constraints, namely perceptual-motor constraints, but at the same time, they are committed to enabling the representation of a comprehensive range of very complex cognitive processes, ranging from multitask performance to problem-solving and learning. Thus, these systems are generally very flexible in what cognitive processes they can represent within their otherwise very constrained architectures.

However, the detail has a downside. Cognitive architectures are typically difficult to program, even for simple tasks, and have the further drawback that, as a consequence of their detail, currently unresolved psychological issues can become exposed to the modeler for resolution. For example, the nature of visual short-term memory is rather poorly understood at this time, and no current architecture has an empirically sound representation of it. Using one of the current architectures to model a task in which visual short-term memory appears to be prominent might require many detailed assumptions about how it works and is used in the task, and these assumptions typically cannot be tested within the modeling project itself. One reason is the difficulty discussed below of getting high-precision data for complex tasks. But the more serious reason is that, in a design context, data to test the model is normally not available because there is not yet a system to collect the data with! Less detailed modeling approaches such as GOMS may not be any more accurate, but they at least have the virtue of not side-tracking the modeler into time-consuming detailed guesswork or speculation about fundamental issues. See Kieras (2005b) for more discussion.

57.2.6.2 Task Networks Can Be Used before Detailed Design

Although model-based evaluation works best for detailed designs, the task network modeling techniques were developed to assist in design stages before detailed design, especially for complex military systems. For example, task network modeling was used to determine how many human operators would be required to properly man a new combat helicopter. Too many operators drastically increase the cost and size of the aircraft; too few means the helicopter could not be operated successfully or safely. Thus, questions at these stages of design are what capacity (in terms of the number of people or machines) is needed to handle the workload and what kinds of work need to be performed by the each person or machine.

In outline, these early design stages involve first selecting a *mission profile*, essentially a high-level scenario that describes what the system and its operators must accomplish in a typical mission, then developing a basic *functional analysis* that determines the functions (large-scale operations) that must be performed to accomplish the mission and what their interactions and dependencies are. Then the candidate high-level design consists of a tentative *function allocation* to determine which human operator or machine will perform each function (see Beevis et al. 1992). The task network model can then be set up to include the tasks and their dependencies, and simulations run to determine execution times and compute workload metrics based on the workload characteristics of each task.

Clearly, entertaining detailed designs for each operator's controls or workstation is pointless until such a high-level analysis determines how many operators there will be and what tasks they are responsible for. Note that the cognitive-architecture and GOMS models are inherently limited to predicting performance in detailed designs, because their basic logic is to use the exact sequence of activities required in a task to determine the sequence of primitive operations. However, as will be discussed later, recent work with *high-level GOMS models* suggests an alternative approach in which a GOMS model using abstract or high-level operators to interact with the device can be developed first and then elaborated into a model for a specific interface as the design takes shape. But at this time, for high-level design modeling, the task-network models appear to be the best, or only, choice.

However, there are limitations that must be clearly understood. The ability of the task network models to represent a design at these earliest stages is a direct consequence of the fact that these modeling methods do not have any detailed mechanisms or constraints for representing human cognition and performance. Recall that the tasks in the network can consist of any arbitrary process whose execution characteristics can follow any desired distribution. Thus, the tasks and their parameters can be freely chosen without any regard to how a human will be actually doing them in the final version of the system. Hence this early-design capability is a result of a lack of theoretical content in the modeling system itself.

While the choice of tasks in a network model is based on a task analysis, the time distribution parameters are more problematic—how does one estimate the time required for a human to perform a process specified only in the most general terms? One way is to rely on empirical measurements of similar tasks performed in similar systems, but this requires that the new system must be similar to a previous system not only at the task-function level, but at least roughly at the level of design details.

Given the difficulty of arriving at task parameter estimates rigorously, a commonly applied technique is to ask a subject matter expert to supply *subjective estimates* of task time means and standard deviations and workload parameters. When used in this way, a task-network model is essentially a mathematically straightforward way to start with estimates of individual subtask performance, with no restrictions on the origin or quality of these estimates and then to combine them to arrive at performance estimates for the entire task and system.

Clearly, basing major design decisions on an aggregation of mere subjective estimates is hardly ideal, but as long as a detailed design or preexisting system is not available, there is really no alternative to guide early design. In the absence of such analyses, system developers would have to choose an early design based on "gut feel" about the entire design, which is surely more dangerous.

Note that if there is a detailed design available, the task-network modeler could decompose the task structure down to a fine enough level to make use of basic human performance parameters, similar to those used in the cognitive-architecture and GOMS models. For example, some commercial tools allow using the HOS micromodels to produce fine-grained task time predictions. However, it is hard to see the advantage in using task network models for detailed design. The networks and their supplementary executable code would seem to be less suitable to the design process than the conceptually simple task procedure descriptions used in GOMS models or the highly flexible and modular structure of production systems current popular in cognitive architectures.

Another option would be to construct GOMS or cognitive architecture models to produce time estimates for the individual tasks and use these in the network model instead of subjective estimates. This might be useful if only part of the design has been detailed, but otherwise, staying with a single modeling approach would surely be simpler. If one believes that interface usability is mostly a matter of getting the details right, along the lines originally argued by Whiteside et al. (1985) and verified by many experiences in user testing, modeling approaches that naturally and conveniently work at a detailed design level will be especially valuable.

57.3 PRACTICAL ISSUES IN APPLYING A MODEL-BASED EVALUATION TECHNIQUE

Once a model-based evaluation technique is chosen, there are some practical issues that arise in seeking to apply the technique to a particular user interface design situation. This section presents several of these issues.

57.3.1 Creating the Simulated Device

As mentioned earlier, the basic structure of a model used for evaluation is that a simulated human representing the user is interacting with a simulated device that represents the system under design. In parallel with Norman's interactive cycle, the simulated human receives simulated visual and auditory input from the simulated device and responds with simulated actions that provide input to the simulated device, which can then respond with different visual and auditory inputs to the human. Depending on the level of generativity and fidelity of the model, the simulated device can range from being a dummy device that does nothing in response to the simulated human interaction to a highly detailed simulation of the device interface and functionality. An example of a dummy device is the device that is assumed in the Keystroke-Level Model, which is not at all explicitly defined; the modeler simply assumes that a specific sequence of user actions will result in the device doing the correct thing. At the other extreme are models such as the ones used by Kieras and Santoro (2004), which actually implemented significant portions of the logical functionality of a complex radar workstation and the domain of moving aircraft and ships. In modeling situations where a generative model is called for, namely a complex task domain with multiple or lengthy detailed scenarios, a fully simulated device is the most convenient way to ensure that a simulated human is in fact performing the task correctly and to easily work with more than one scenario for the task situation.

It is important to realize that the simulated device does not have to implement the actual interface whose design is being evaluated. Rather, it suffices to produce abstract psychological inputs to the simulated human. For example, if a red circle is supposed to appear on the screen, the simulated device can merely signal the simulated human with an abstract description that an object has appeared at certain (x, y) coordinates that has a "shape" of "circle" and a "color" of "red." It is not at all necessary for the simulated device to actually produce a human-viewable graphical display containing a circular red area at a certain position.

A lesson learned by Kieras and Santoro (2004) was that the effort required to construct even such an abstract simulated device in a complex domain is a major part of the modeling effort and is more difficult in some ways than constructing the models of the simulated users! Clearly, to some extent, this effort is redundant with the effort required to develop the actual system and so can undermine the rationale for modeling early in the design and development process.

A common response to this problem is to seek to connect the cognitive architecture directly to an intact application or system prototype that plays the role of the simulated device, in short, replacing the simulated device with an actual device. Work such as St. Amant and Reidl (2001) provides a pathway for interfacing to an intact application: the technique is basically to use the existing API of the application platform (e.g. Windows) to capture the screen bitmap and run visual object recognition algorithms on it to produce the description of the visual inputs to the simulated human, and outputs from the simulated human can be directly supplied to the platform to produce keyboard input or to control the cursor position. Even for the limited domain of Windows applications using the standard GUI objects, this is technically challenging, but quite feasible.

A less ambitious approach is to instrument a prototype version of the interface, so that for example, when the prototype causes a certain object to appear on the screen, the simulated human is supplied with the visual input description. Given the considerable variety in how GUIs are implemented, this solution is not very general, but does have interesting solutions if the application prototype is programmed in Java, HTML, or similar cross-platform languages or general-purpose tools that can be used for prototyping.

However, both these methods of coupling a user model to an application suffer from an easily overlooked limitation: The time when modeling is most useful is *early* in design, *before* the system has been prototyped. Thus, because coupling to a prototype or an application can only happen late in the development process or after development, these approaches come too late to provide the most benefit of model-based evaluation.

Thus, multiple approaches for creating the simulated device are both possible and needed: if the design questions can be answered with evaluation techniques such as the Keystroke-Level Model, then no simulated device is needed at all. If the model is for an existing application, coupling to the intact application is clearly the best solution. If a prototype is going to be constructed at this point in the design process anyway, using it as the simulated device is the best solution. But in the potentially most useful case, the simulated device must be created before any prototype or final application, making the fewest possible commitments to prototyping or coding effort; this requires constructing a simulated device from scratch, stripped down to the bare minimum necessary to allow a candidate design to interact with the simulated user. The next section provides some advice on this process.

57.3.1.1 How to Simplify the Simulated Device

Distinguish between device behavior that is relevant to the modeling effort and that which is not. Basically, if the simulated human will not use or respond to certain behaviors of the simulated device, then the simulated device does not need to produce those behaviors. A similar argument applies to the amount of detail in the behavior. Of course, as the interface design is elaborated, the simulated device may need to cover more aspects of the task. Good programming techniques will make it easy to extend the simulated device as needed; a good programmer on the project is a definite asset.

Distinguish between what the simulated device has to provide to the simulated human and what would be a convenience to the modeler. That is, while the device can supply abstract descriptions to the simulated user, an actual graphical display of what the simulated device is displaying can be a very useful tool for the modeler in monitoring, debugging, or demonstrating the model. A very crude general-purpose display module that shows what the simulated human "sees" will suffice and can be reasonably easy to provide in a form that is reusable for a variety of simulation projects. However, developing this handy display should be recognized as an optional convenience, rather than an essential part of the simulated device.

Since programming the simulated device can be a significant programming effort, an attractive simplification would be a programming language that is specialized for describing abstract device behavior. Clearly, using such a language could be valuable if the modeling system already provides it and it is adequate for the purpose, especially if the device programming language can generate a prototype for the interface that can be directly coupled to the simulated human, moving the whole process along rapidly. An extensive project involving many different but similar interface designs would profit, especially if the language matches the problem domain well.

However, in less than ideal situations, a specialized device language is unlikely to be an advantage. The reason is that to cover the full span of devices that might need to be simulated, the device programming language will have to include a full set of general programming language facilities. For example, to handle the Kieras and Santoro (2004) domain, trigonometric functions are needed to calculate courses and trajectories and containers of complex objects are required to keep track of the separate aircraft and their properties. Thus, specialized languages will inevitably have to include most of the same feature set as general-purpose programming languages, meaning that the developers of modeling systems will have to develop, document, maintain, and support with editors and debuggers a full-fledged programming language. This takes effort away from the functions that are unique to human performance modeling systems, such as ensuring that the psychology is correctly represented. In addition, the modeler will also have to expend the time and effort necessary to learn a specialized language whose complexity is similar to a general-purpose programming language, also taking effort away from unique aspects of the modeling effort.

A better choice would be to provide for the device to be programmed easily in a standard general-purpose programming language that modelers can (or should) know anyway, allowing reuse of not just the modeler's skills, but existing programming tools and education resources as well. A well-designed modeling system can ensure that a minimum of system-specific knowledge must be acquired before coding can begin.

57.3.2 Identifying the Task Strategy

57.3.2.1 Task Analysis Does Not Necessarily Specify a Task Strategy

Human performance in a task is determined by (1) the logical requirements of the task—what the human is supposed to accomplish, as determined by a task analysis; (2) the human cognitive architecture—the basic mechanisms available to produce behavior; and (3) a specific strategy for doing the task—given the task requirements and the architecture, what should be done in what order and at what time to complete the task. Thus, to construct a model for doing the task, one must first understand the task, then choose an architecture, and then choose a strategy to specify how the architecture will be used to accomplish the task. Identifying this strategy is the critical prerequisite for constructing a model.

Normal task analysis methods, such as those described in sources such as Kirwan and Ainsworth (1992), Beevis et al. (1992), and Diaper and Stanton (2004), do not necessarily identify the exact sequence of actions to perform with the interface under design, and they rarely specify the timing of actions. For example, anyone who has made coffee with a home coffee maker knows that there are certain constraints that must be met, but there is still considerable variation in the sequence and time in which the individual required steps could be performed. In fact, there can be variation on how the activity is organized; for example, one strategy is to use the visible state of the coffee maker as an external memory to determine which actions should be performed next (Larkin 1989). A normal task analysis will not identify these variations. But even further, task analysis will not necessarily identify how any trade-offs should be decided, even as basic as speed versus accuracy, much less more global problems such as managing workload, dealing with multiple task priorities, and so forth.

Thus, to model a human performing in such situations, some additional information beyond a normal task analysis has to be added, namely the specific task strategies that are used to accomplish the tasks. Conversely, the performance that a human can produce in a task can vary over a wide range depending on the specific strategy that is used, and this is true over the range of tasks from elementary psychology laboratory tasks to highly complex real-world tasks (Kieras and Meyer 2000). This raises a general problem: Given that we have a model that predicts performance on an interface design, how do we distinguish the effects of the interface design from the effects of the particular strategy for using the interface? Not only does this apply to the model performance, but also to the human's performance. It has always been clear that clever and experienced users can get a lot out of a poorly designed system, and even a reasonably well-designed powerful system can be seriously under-used (Bhavnani and John 1996). How can we predict performance without knowing the actual task strategy, and how does our model's task strategy relate to the actual user's task strategy?

57.3.2.2 Difficulties in Identifying Task Strategy

The state of the art in cognitive modeling research for identifying a task strategy is to choose a candidate intuitively, build the model using the strategy, evaluate the goodness of fit to data, and then choose a better strategy and repeat until a satisfactory fit is obtained. If there is adequate detail in the data, such as the sequence of activities, it might be possible to make good initial guesses at the task strategy and then revise these through the modeling process. This iterative refinement process is known to be very slow, but more seriously, in system design we normally do not have data to fit a model to—this is what the modeling is supposed to replace. The task strategy has to be chosen in the absence of such data.

Another approach is to get the task strategy by knowledge engineering techniques with existing task performers or other sources such as training materials. As will be argued below, it is especially important to identify the best (or at least a good) strategy for doing the task. But a good strategy for doing a task is often not obvious, even to experts. Even highly experienced people do not always know or use the best procedures; even the trainers may not know them, and it is common to discover that procedural training materials present suboptimal methods. Finally, and again most importantly, if the system is new, there are no experts or training materials to consult to see how it is used.

57.3.2.3 A Heuristic: Model What Users Should Do

Given the obstacles to identifying task strategies, how do we find out what strategies users will follow in using the system under development? The short answer is that it is too hard to find out within the constraints of a design process. Instead, start from the design goals that the system is supposed to meet and assume that *users will be using the system like it is supposed to be used*. For example, if the system provides a feature that is supposed to allow the user to perform a certain task easily, assume that the simulated user will use that feature in the intended fashion. This is essentially a best-case analysis of the ability of the user to make use of the interface design. If the usability under these conditions is too low, then it would certainly be inadequate when used by actual users! It is a separate issue whether the users can or will use the system in this intended way and the failures can be dramatic and serious (Bhavnani and John 1996). Whether users use a system in the intended way depends on several factors: problems in the learnability of the design, which some models (see John and Kieras 1996a,b) can predict; the quality of the training materials, which also can be improved with modeling (see John and Kieras 1996a,b); and perhaps most importantly, matters beyond the scope of model-based evaluation, such as whether users have the opportunity or incentive to take full advantage of the system. Finally, there is no point in trying to improve a system under the assumption that users will ignore the capabilities that it provides! This is why, in choosing a task strategy to model for design evaluation, the most effective approach is to assume that *the design will be used as intended*. Not only will this strategy be the easiest to identify and implement, but it is also most directly relevant to evaluating the design!

Within this basic strategy, there is another range of variation, which is whether the user attempts to perform at the highest possible level or simply satisfices by performing at some adequate level. That is, people can use clever low-level strategies (what Gray and Boehm-Davis [2000] termed microstrategies) to greatly improve performance. For example, if the task is to classify "blips" on a radar display, the user can speed up considerably by looking for the next blip to inspect while still hitting the keys to respond to the previous one (Hornof, Zhang, and Halverson 2010). On the other hand, the user is performing reasonably if they finish each blip before going on to the next one. Kieras and Meyer (2000) pointed out that in a variety of even elementary laboratory tasks, subjects do not always adopt high-performance strategies, even when large improvements would result; they are optional, not mandatory. So even if we are willing to assume that the design will be used as intended, how do we know whether the actual users will be going "all out" with a clever strategy versus just "getting the job done?" Again, the short answer is that it is too difficult to find out, especially in a design process where the system does not yet exist.

In response to this quandary, Kieras and Meyer (2000) proposed the *bracketing heuristic*. Construct a base model in which the user performs the task in a straightforward way using the interface as designed, but without any special strategic optimizations, a *slowest-reasonable* model. Derive from this a *fastest-possible* model that performs the task at the maximum performance level allowed by the cognitive architecture used for the model. The two models should bracket the actual user's future performance. If both models produce adequate performance, then the design should be adequate; if both produce inadequate performance, then the design needs to be improved. If the slowest-reasonable model is inadequate, but the fastest-possible model is acceptable, boosting the level of training or perhaps motivation of the user might result in satisfactory performance, although clearly improving the design would be a more robust solution.

There has been recent work devoted to automatic construction of the optimum strategy corresponding to the fastest-possible model. The concept is that a cognitive architecture and task demands can be specified as a set of constraints. Procedures or task strategies can then be computationally generated from these constraints and their performance compared (John et al. 2002; Vera et al. 2004; Howes, Lewis, and Vera 2009). Thus, it is possible to algorithmically explore the space of all possible strategies and then choose the strategy that maximizes task performance. This approach takes much of the guess-work out of model-building, at least for the fastest/best-possible model. While very promising, it is not yet clear whether the technique can be practically scaled to evaluate realistically complex interfaces and tasks, or whether the slowest-reasonable model or its equivalent can be similarly automatically constructed.

57.3.3 Concerns over Model Validity

57.3.3.1 Can You Believe the Model?

Suppose a model implies critical design choices. Should you follow them? A poor response is to build and test prototypes just as if no modeling had been done. It could be argued that the modeling might have clarified the situation, but the purpose of model-based evaluation is to reduce the amount of prototyping and user testing required for refining a design. So this response under-utilizes the approach.

A better response to the situation is to understand how the model implies the design choices—what aspects of the model are contributing to the outcome? This can be done by profiling the model processing and analyzing the model structure. If the critical aspects of the model are known to be valid and appear to be properly represented, then the model results should be accepted. For example, perhaps one design is slower than the other simply because it turns out that more navigation through menus is required; the model processes involved are relatively simple and adequately validated in the literature. However, if the relevant aspects of the model are problematic, the result needs further study. For example, suppose the model for the better of two designs differs from the poorer design in assuming that the user can remember all the information about previously inspected screen objects and so does not need to search the screen again when the information is needed later. Because the bounds on visual memory are unclear, as discussed above, the modeling architecture might not enforce any bounds. Thus, the modeling result is suspicious for this reason, and the true bounds might be much smaller than the model assumes. The modeler could then perform a sensitivity analysis to reveal how much the design choice might be affected by the problematic assumption. For example, the model could be modified to vary the number of previous inspected objects that could be remembered. At what value does this parameter change which design is predicted to be better? If the decision is not very sensitive to this parameter (e.g., because the effects are minor compared to the total time involved or there are other improvements in the better design), then choosing the better design is a reasonably safe decision.

If the decision turns out to be sensitive to the problematic model assumption, then the situation becomes quite difficult. One possible solution is to remove the problematic aspect of the model by changing it in the direction of less human capability; this is a conservative strategy and might be appropriate if the user will be under stress as well. But if data or theory that resolves the issue is available, the modeler can go beyond the normal model-based process and modify the model or architecture to incorporate the more accurate psychological information.

57.3.3.2 Should You Validate the Model?

Remember that testing with real users must be done sometime in the development process, because the models do not cover all the design issues, are only approximate where they do apply, and like any analytic method, can be misapplied.

The model can thus be validated after use by comparing the final user test results to the model predictions; this will reveal problems in the accuracy of the model and its application to the design; any modeling mistakes and design errors can then be corrected for the future.

However, should special data to validate the model be collected prior to using it to guide the design? While it would seem to be a good idea to validate a model before using it, the answer really should be *no* because validation is not supposed to be a normal part of using a predictive model of human performance. The whole idea of model-based evaluation is to avoid data collection during design. Only the developers of the modeling methodology are supposed to be concerned about the validity, not the user of the methodology.

There are a couple of special cases about data collection that need discussion. One is data collection to provide basic parameter values required for modeling, such as how long it takes the user to input characters with a novel device. If the parameters concern low-level processes, the data collection is independent of a specific design and will be generally useful for many different modeling applications. The second special case is data collection to support modeling how an *existing* system is *actually* being used. Such a model cannot be constructed a priori, but rather must be based on data about how actual users interact with the actual system. Because of the uncertainties involved in constructing a model based on human behavior, the model will have to be validated with a suitably complete and reliable data set before it can be taken as a usefully accurate model. This purpose of modeling is very different from the model-based evaluation approach presented in this chapter: instead of serving as a guide for designing a new system and a surrogate for user testing, the model is an explanation and characterization of observed behavior; it might serve as a guide for a new design, but only in the sense of characterizing the current situation that we want to improve upon. The model itself will not directly apply to the new design. In short, modeling the actual use of an existing system has very different methods, goals, and applications from modeling the usage of a system being designed.

But if in spite of all the above considerations, the validity of the model is in question, it is critical to realize that a data set adequate for a scientifically sound test of validity must be much more controlled and detailed than normal user testing data and can be very difficult to collect in the context of a development project, due to the practical difficulties that appear in both applied and basic research. Despite the usually considerable effort and expense to collect it, data on actual real-world task performance is often lacking in detail, scenario coverage, and adequate sample sizes. Even in the laboratory, collecting highly precise, detailed, and complete data about task performance is quite difficult, and researchers are typically trapped into using tasks that are artificial, performed by nonexpert subjects, or trivial relative to actual tasks, and even so, the data collection, analysis, and interpretation process can take years to complete.

To elaborate on the difficulty of collecting adequate validation data, while all would agree that a model is almost certainly incorrect at some level, it is often mistakenly assumed that collecting empirical data on human performance in complex tasks is a royal road to certainty. Rather, as pointed out in Kieras and Santoro (2004), complex real-world tasks involve subtle user strategies, team interactions, influences of background knowledge, and the specifics of the scenarios. Such experiments are extremely slow and expensive to conduct, even with small samples, where the reliability of the results then comes into question. Clearly, it is not practical to run experiments using many scenarios, every reasonable design variation, every candidate team organization, and ample numbers of subjects. Furthermore, even for a well-chosen subset of these possibilities, it may be difficult to understand why people did what they did in the tasks—asking them is usually ambiguous at best and their strategies might be idiosyncratic. Thus, the reliability, generalizability, and even the meaning of the data can be difficult to determine. In fact, it can be difficult to ensure that the model and the experiment are even relevant to each other. For example (see Kieras and Santoro 2004), if the model is based on what users *should* do in the task and the test users do not follow the strategy that the model follows, then the failure of the model to behave the same way as the test users is actually irrelevant—it might be said that the data is "wrong," not the model! Thus, even if deemed appropriate, attempting to validate a model of a complex task is likely to be impractically difficult for a normal development process.

A final point on data collection is as follows: if the resources are available to do extensive data collection and analysis before the final stages of development, what function is served by modeling? If the model validity would be in doubt, would not the data collection resources be better devoted to user testing?

57.3.3.3 Summary: Assessing Model Validity

Instead of collecting data to validate the model, assess its validity in terms of whether it meets the following basic requirements: (1) Is the model strategy based on an analysis of what users *should* do? If not, it is a poor choice to inform the design of a new system. (2) Is it likely that users can or will follow the same strategy as the model? If not, then the model is irrelevant—either the model was misconstructed or the design is fundamentally wrong. (3) Are the assumptions about human abilities in the model plausible? If not, see the earlier suggestions. If the answer to all three questions is yes, then the model results can be accepted as useful guidance for the design decisions without special validation efforts. Of course, it needs to be kept in mind that the model might be seriously incorrect, but modeling is not supposed to be perfect; it suffices merely that it help a design process.

57.4 GOMS MODELS: READY-TO-USE APPROACH

As summarized earlier, GOMS is an approach to describing the knowledge of procedures that a user must have to operate a system. The different types of GOMS models differ in the specifics of how the methods and sequences of operators are represented. The aforementioned CPM-GOMS model represents a specific sequence of activity in terms of the cognitive, perceptual, and motor operators performed in the context of a simple model of human information processing. At the other extreme of detail, the Keystroke-Level Model (Card, Moran, and Newell 1980) is likewise based on a specific sequence of activities, but these are limited to the overt *keystroke-level* operators (i.e., easily observable actions at the level of keystrokes, mouse moves, finding something on the screen, turning a page, and so forth). The task execution time can be predicted by simply looking up a standardized time estimate for each operator and then summing the times. The Keystroke-Level Model has a long string of successes to its credit (see John and Kieras 1996a). Without a doubt, if the design question involves which alternative design is faster in fairly simple situations, there is no excuse for measuring or guessing when a few simple calculations will produce a usefully accurate answer.

However, the Keystroke-Level Model is tedious to calculate for realistic applications and requires training and practice to apply uniformly, and in its bare form does not fit well into traditional user interface development processes. A good solution, CogTool, has been developed by John and co-workers (John et al. 2004) (see http://cogtool.hcii.cs.cmu.edu/). CogTool takes advantage of the *storyboard* technique, the most common interface design technique in which illustrations of the sequence of displays show how the user moves through a task. These storyboards can be in the form of simple hand-drawn sketches or more elaborate mockups in the form of PowerPoint slides or HTML pages. The tool allows the user to bring in these storyboard images, annotate them with "hot-spots" for the buttons if needed, and then easily demonstrate the sequence of user actions that accomplish the task. The tool then automatically calculates the Keystroke-Level Model time predictions for each task. One result is that the tool greatly improves the reliability of the modeling—different tool users produce very similar modeling results (John 2010). But more importantly, CogTool provides a simple and effective storyboard-level prototyping tool that also provides basic task time predictions as a "bonus," making it much more tempting to the developer community. This approach is an important step in making model-based evaluation more practical and accepted.

It is easy to generalize the Keystroke-Level Model somewhat to apply to more than one specific sequence of operators. For example, if the scenario calls for typing in some variable number of strings of text, the model can be parameterized by the number of strings and their length. However, if the situation calls for complex branching or iteration and clearly involves some kind of hierarchy of task procedures, such sequence-based models become quite awkward and a more generative form of model is required.

The generative forms of GOMS models are those in which the procedural knowledge is represented in a form resembling an ordinary computer programming language and are written in a fairly general sort of way. This form of GOMS model can be applied to many conventional desktop computing interface design situations. It was originally presented in Card, Moran, and Newell (1983, Chapter 5) and further developed by Kieras, Polson, and Bovair (Kieras and Polson 1985; Polson 1987; Bovair, Kieras, and Polson 1990), who provided a translation between GOMS models and the production-rule representations popular in several cognitive architectures and demonstrated how these models could be used to predict learning and execution times. Kieras (1988, 1997) proposed a structured-natural-language notation, NGOMSL ("Natural" GOMS language), which preserved the empirical content of the production-rule representation, but resembled a conventional procedural programming language. This notation was later formalized into a fully executable form, GOMSL (GOMS language), for use in computer simulation tools that implement a simplified cognitive architecture (Kieras et al. 1995; Kieras 2005a). This tool has been applied to modeling team tasks (e.g. Kieras and Santoro 2004) and extended to provide analysis of error recovery methods supported by error-source heuristics (Wood 2000). St. Amant, Horton, and Ritter (2007) provide a useful comparison of a GOMS model with a cognitive-architectural model, showing how the GOMS model can provide results comparable to the more complex model much more easily.

Continuing the analogy with conventional computer programming languages, in generative GOMS models, the operators are like the primitive operations in a programming language; methods are like functions or subroutines that are called to accomplish a particular goal, with individual steps or statements containing the operators, which are executed one at a time, as in a conventional programming language. Methods can assert a sub-goal, which amounts to a call of a sub-method, in a conventional hierarchical flow of control. When procedural knowledge is represented explicitly in this way and in a format that enforces a uniform "grain size" of the operators and steps in a method, then there are characteristics of the representation that relate to usability metrics in straightforward ways.

For example, the collection of methods represents "how to use the system" to accomplish goals. If a system requires a large number of lengthy methods, then it will be hard to learn; there is literally more required knowledge than for a system with a smaller number of methods or simpler methods. If the methods for similar goals are similar, or in fact the same method can be used to accomplish different, but similar, goals, then the system is "consistent" in a certain, easily characterized sense: in a procedurally consistent system, fewer methods, or unique steps in methods, must be learned to cover a set of goals compared to an inconsistent system, and so it is easier to learn. One can literally count the amount of overlap between the methods to measure procedural consistency.

Finally, by starting with a goal and the information about the specific situation, one can follow the sequence of operators specified by the methods and sub-methods to accomplish the goal. This generates the sequence of operators required to accomplish the goal under that specific situation; if the methods were written to be adequately general, they should suffice to generate the correct sequence of operators for any relevant task situation. The times for the operators in the trace can be summed, as in the Keystroke-Level Model, to obtain a predicted execution time. Details of the timing can be examined or "profiled" to see where the processing bottlenecks are.

See Baumeister, John, and Byrne (2000) for a survey of computer tools for GOMS modeling.

57.4.1 Why GOMS Models Work

The reasons why GOMS models have useful predictive and heuristic power in interface design can be summarized under three principles: The *rationality principle* (cf. Card, Moran, and Newell 1983) asserts that humans attempt to be efficient given the constraints on their knowledge, ability, and the task situation. Generally, when people attempt to accomplish a goal with a computer system, they do not engage in behavior that they know is irrelevant or superfluous—they are focused on getting the job done. Although they might perform suboptimally due to poor training (see Bhavnani and John 1996), they generally try to work as efficiently as they know how, given the system they are working with. How they accomplish a goal depends on the design of the system and its interface—for example, in a word-processing system, there are only a certain number of sensible ways to delete a word and the user has some basis for choosing between these that minimizes effort along some dimension. Between these two sets of constraints—the user's desire to get the job done easily and efficiently and the computer system's design—there is considerable constraint on the possible user actions. This means that we can predict user behavior and performance at a useful level of accuracy *just from the design of the system and an analysis of the user's task goals and situation*. A GOMS model is one way of combining this information to produce predicted performance.

Procedural primacy is the claim that regardless of what else is involved in using a system, at some level the user must infer, learn, and execute procedures to accomplish goals using the system. That is, computers are not used purely passively—the user has to do something with them, and this activity takes the form of a procedure that the user must acquire and execute. Note that even display-only systems still require some procedural knowledge for visual search—for example, making use of the flight status displays at an airport requires choosing and following some procedure for finding one's flight and extracting the desired information—different airlines use different display organizations, some of which are probably more usable than others. Because the user must always acquire and follow procedures, the complexity of the procedures entailed by an interface design is therefore related to the difficulty of using the interface. While other aspects of usability are important, the procedural aspect is always present. Therefore, analyzing the procedural requirements of an interface design with a technique such as GOMS will provide critical information on the usability of the design.

Explicit representation refers to the fact that any attempt to assess something benefits from being explicit and clear and relying on some form of written formalized expression. Thus, all task analysis techniques (Kirwan and Ainsworth 1992; Beevis et al. 1992; Diaper and Stanton 2004) involve formalized descriptions of a user's task. Likewise, capturing the procedural implications of an interface design will benefit from representing the procedures explicitly in a form that allows them to be inspected and manipulated. Because GOMS models involve writing out user procedures in a complete, accurate, and detailed format, it becomes possible to define metrics over the representation (e.g., counting the number of statements) that can be calibrated against empirical measurements to provide predictions of usability. Moreover, by making user procedures explicit, the designer can then apply the same kinds of intuition and heuristics used in the design of software: clumsy, convoluted, inconsistent, and "ugly" user procedures can often be spotted and corrected just like poorly written computer code. Thus, by writing out user procedures in a notation like GOMS, the designer can often detect and correct usability problems without even performing the calculations.

A related developing concept that uses an explicit representation of user procedures is the application of *model-checking* methodology developed in computer science for formal verification. Briefly, if the possible states and state transitions of a system can be formally characterized, then it is possible to enumerate all the possible transitions in a system and verify that, for example, the system will not enter into an unsafe state. This has begun to be applied to systems with user interfaces (e.g., Bolton and Bass 2009) to verify their efficacy and safety. While this method does not provide a way for predicting task performance, it may provide a way for verifying the overall functionality and safety of a system.

User procedure representations derived from a task analysis can help choose the functionality behind the interface. The approach can be formalized with *high-level GOMS models* (Kieras 2004), in which the operators refer to invocations of system functions rather than keystroke-level interface actions. GOMSL and GLEAN (see Kieras 2005b) currently contain high-level GOMS operators to directly support a "seamless" transition of computational modeling from the task and functionality level of analysis down to detailed design. Butler et al. (2007) argue that the *top-level routine* that describes how a user and machine cooperate to get work done is usually not deliberately designed, but rather is left up to users to infer from haphazardly chosen functionality. The top-level routine, and the choice of functionality to simplify it and make it more effective, can in fact be designed using techniques based on task and domain analysis, and candidates can be compared and evaluated with high-level GOMS models and potentially verified with model-checking. This combination of techniques would comprise a valuable engineering toolbox for system design.

57.4.2 Limitations of GOMS Models

GOMS models address only the procedural aspects of a computer interface design. This means that they do not address a variety of nonprocedural aspects of usability, such as the readability of displayed text, the discriminability of color codes, or memorability of command strings. Fortunately, these properties of usability are directly addressed by standard methods in human factors.

Within the procedural aspect, user activity can be divided into the open-ended "creative" parts of the task, such as composing the content of a document or thinking of the concept for the design of an electronic circuit, on the one hand, and the routine parts of the task, on the other, which consist of simply manipulating the computer to accept the information that the user has created and then to supply new information that the user needs. For example, the creator of a document has to input specific strings of words in the computer, rearrange them, format them, spell check them, and then print them out. The creator of an electronic device design has to specify the circuit and its components to a CAD system and then obtain measures of its performance. If the user is reasonably skilled, these activities take the form of executing routine procedures involving little or no creativity.

The bulk of time spent working with a computer is in this routine activity, and the goal of computer system design should be to minimize the difficulty and time cost of this routine activity so as to free up time and energy for the creative activity. GOMS models are easy to construct for the routine parts of a task, because, as described earlier, the user's procedures are constrained by the task requirements and the design of the system, and these models can then be used to improve the ability of the system to support the user. However, the creative parts of task activity are purely cognitive tasks and, as discussed above, attempting to formulate a GOMS model for them is highly speculative at best and would generally be impractical. Applying GOMS thus takes some task analysis skill to identify and separate the creative and routine procedural portions of the user's overall task situation.

Finally, it is important to recognize that while a GOMS model is often a useful way to express the results of a task analysis, similar to the popular HTA technique (Annett et al. 1971; Kirwan and Ainsworth 1992), building a GOMS model does not "do" a task analysis. The designer must first engage in task analysis work to understand the user's task before a GOMS model for the task can be constructed. In particular, identifying the top-level goals of the user and selecting relevant task scenarios are all logically prior to constructing a GOMS model.

57.5 CONCLUDING RECOMMENDATIONS

- If you need to predict the performance of a system prior to detailed design when overall system structure and functions are being considered, use a task network model.
- If you are developing a detailed design and want immediate intuitive feedback on how well it supports the user's tasks, write out and inspect a high-level or informal GOMS model for the user procedures while you are making the design decisions.
- If your design criterion is the execution speed for a discrete selected task, use a Keystroke-Level model.
- If your design criteria include the learnability, consistency, or execution speed of a whole set of task procedures, use a generative GOMS model. If numerous or complex task scenarios must be modeled, use a GOMS model simulation system.
- If the design issues hinge on understanding detailed or subtle interactions of human cognitive, perceptual, and motor processing and their effect on execution speed and only a few scenarios need to be analyzed, use a CPM-GOMS model.
- If the resources for a research-level activity are available and a detailed analysis is needed of the cognitive, perceptual, and motor interactions for a complex task or many task scenarios, use a model built with the simplest cognitive architecture that incorporates the relevant scientific phenomena.

REFERENCES

Anderson, J. R. 1983. *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.

Annett, J., K. D. Duncan, R. B. Stammers, and M. J. Gray. 1971. *Task Analysis*. London: Her Majesty's Stationery Office.

Baumeister, L. K., B. E. John, and M. D. Byrne. 2000. A comparison of tools for building GOMS models. In *Proceedings of CHI 2000*. New York: ACM.

Beckett, D. 1984. *Stephenson's Britain*. North Pomfret, Vermont: David and Charles, Inc.

Beevis, D., R. Bost, B. Doering, E. Nordo, F. Oberman, J.-P. Papin, I. H. Schuffel, and D. Streets. 1992. *Analysis Techniques for Man-Machine System Design*. (Report AC/243(P8)TR/7). Brussels, Belgium: Defense Research Group, NATO HQ.

Bhavnani, S. K., and B. E. John. 1996. Exploring the unrealized potential of computer-aided drafting. In *Proceedings of the CHI '96 Conference on Human Factors in Computing Systems*. New York: ACM.

Bolton, M. L., and E. J. Bass. 2009. A method for the formal verification of human-interactive systems. In *Proceeding of the Human Factors and Ergonomics Society 53rd Annual Meeting*. 764–8. Santa Monica, CA: Human Factors and Ergonomics Society.

Bovair, S., D. E. Kieras, and P. G. Polson. 1990. The acquisition and performance of text editing skill: A cognitive complexity analysis. *Hum Comput Interact* 5:1–48.

Buchanan, R. A. 1989. *The Engineers: A History of the Engineering Profession in Britain 1750-1914*. London: Jessica Kingsley Publishers.

Butler, K. E., J. Zhang, C. Esposito, A. Bahrami, R. Hebron, and D. Kieras. 2007. Work-centered design: A case study of a mixed-initiative scheduler. In *Proceedings of CHI 2007*. San Jose, California, New York: ACM.

Card, S. K., T. P. Moran, and A. Newell. 1980. The keystroke-level model for user performance time with interactive systems. *Commun ACM* 23(7):396–410.

Card, S., T. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, New Jersey: Erlbaum.

Chubb, G. P. 1981. SAINT, a digital simulation language for the study of manned systems. In *Manned System Design*, eds. J. Moraal, and K. F. Kraas, 153–79. New York: Plenum.

Diaper, D., and N. A. Stanton, eds. 2004. *The Handbook of Task Analysis for Human-Computer Interaction*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Elkind, J. I., S. K. Card, J. Hochberg, and B. M. Huey, eds. 1989. *Human Performance Models for Computer-Aided Engineering*. Committee on Human Factors, National Research Council. Washington: National Academy Press.

Glenn, F. A., A. L. Zaklad, and R. J. Wherry. 1982. Human operator simulation in the cognitive domain. In *Proceedings of the Human Factors Society*, 964–9. Santa Monica, CA: Human Factors and Ergonomics Society.

Gordon, J. E. 1978. *Structures: Or Why Things Don't Fall Down.* New York: Plenum.

Gray, W. D., and D. A. Boehm-Davis. 2000. Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *J Exp Psychol Appl* 6(4):322–35.

Gray, W. D., B. E. John, and M. E. Atwood. 1993. Project Ernestine: A validation of GOMS for prediction and explanation of real-world task performance. *Hum Comput Interact* 8(3):237–09.

Harris, R. M., H. P. Iavecchia, and A. C. Bittner. 1988. Everything you always wanted to know about HOS micromodels but were afraid to ask. In *Proceedings of the Human Factors Society*, 1051–5. Santa Monica, CA: Human Factors and Ergonomics Society.

Harris, R., H. P. Iavecchia, and A. O. Dick. 1989. The human operator simulator (HOS-IV). In *Applications of Human Performance Models to System Design*, eds. G. R. McMillan, D. Beevis, E. Salas, M. H. Strub, R. Sutton, and L. Van Breda, 275–80. New York: Plenum Press.

Hornof, A. J., Y. Zhang, T. Halverson. 2010. Knowing where and when to look in a time-critical multimodal dual task. In *Proceedings of ACM CHI 2010: Conference on Human Factors in Computing Systems*, 2103–12. New York: ACM.

Howes, A., R. L. Lewis, and A. Vera. 2009. Rational adaptation under task and processing constraints: Implications for testing theories of cognition and action. *Psychol Rev* 116(4):717–51.

John, B. E. 2010. Reducing the variability between novice modelers: Results of a tool for human performance modeling produced through human-centered design. In *Proceedings of the 19th Annual Conference on Behavior Representation in Modeling and Simulation (BRIMS)*. Orlando, FL: SISO, Inc.

John, B. E., and D. E. Kieras. 1996a. Using GOMS for user interface design and evaluation: Which technique? *ACM Trans Comput Hum Interact* 3:287–319.

John, B. E., and D. E. Kieras. 1996b. The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Trans Comput Hum Interact* 3:320–51.

John, B., K. Prevas, D. Salvucci, and K. Koedinger. 2004. Predictive human performance modeling made easy. In *Proceedings of CHI 2004.* New York: ACM.

John, B., A. Vera, M. Matessa, M. Freed, and R. Remington. 2002. Automating CPM-GOMS. In *Proceedings of CHI 2002*. New York: ACM.

Kieras, D. E. 1988. Towards a practical GOMS model methodology for user interface design. In *Handbook of Human–Computer Interaction*, ed. M. Helander, 135–58. Amsterdam: North–Holland Elsevier.

Kieras, D. E. 1997. A Guide to GOMS model usability evaluation using NGOMSL. In *Handbook of Human-Computer Interaction*. eds. M. Helander, T. Landauer, and P. Prabhu, Second ed. 733–66. Amsterdam: North-Holland.

Kieras, D. E. 2004. Task analysis and the design of functionality. In *The Computer Science and Engineering Handbook*. ed. A. Tucker, 2nd ed. 46-1–46-25. Boca Raton, FL: CRC Inc.

Kieras, D. E. 2005a. *A Guide to GOMS Model Usability Evaluation using GOMSL and GLEAN4.* Document available at http://www.eecs.umich.edu/~kieras/ (accessed November 30, 2011).

Kieras, D. 2005b. Fidelity issues in cognitive architectures for HCI modeling: Be careful what you wish for. Paper presented at the 11th International Conference on Human Computer Interaction (HCII 2005). Las Vegas, July 22–27. Proceedings published as CD-ROM.

Kieras, D. 2009. Why EPIC was wrong about motor feature programming. In *9th International Conference on Cognitive Modeling—ICCM 2009*, eds. A. Howes, D. Peebles, and R. Cooper. Manchester, U.K. Available online at http://sideshow.psyc.bbk.ac.uk/rcooper/iccm2009/proceedings/ (Retrieved November 30, 2011).

Kieras, D. E., and D. E. Meyer. 2000. The role of cognitive task analysis in the application of predictive models of human performance. In *Cognitive Task Analysis*, eds. J. M. C. Schraagen, S. E. Chipman, and V. L. Shalin, 237–60. Mahwah, NJ: Lawrence Erlbaum.

Kieras, D. E., and P. G. Polson. 1985. An approach to the formal analysis of user complexity. *Int J Man Mach Stud* 22:365–94.

Kieras, D. E., and T. P. Santoro. 2004. Computational GOMS modeling of a complex team task: Lessons learned. In *Proceedings of CHI 2004: Human Factors in Computing Systems*. New York: ACM, Inc.

Kieras, D. E., S. D. Wood, K. Abotel, and A. Hornof. 1995. GLEAN: A computer-based tool for rapid GOMS model usability evaluation of user interface designs. In *Proceeding of UIST*, 91–100. Pittsburgh, PA, New York: ACM.

Kieras, D. E., S. D. Wood, and D. E. Meyer. 1997. Predictive engineering models based on the EPIC architecture for a multimodal high-performance human-computer interaction task. ACM *Trans Comput Hum Interact* 4:230–75.

Kirwan, B., and L. K. Ainsworth. 1992. *A Guide to Task Analysis*. London: Taylor & Francis.

Landauer, T. 1995. *The Trouble with Computers: Usefulness, Usability, and Productivity.* Cambridge, MA: MIT Press.

Lane, N. E., M. I. Strieb, F. A. Glenn, and R. J. Wherry. 1981. The human operator simulator: An overview. In *Manned System Design*, eds. J. Moraal and K. F. Kraas, 121–52. New York: Plenum.

Larkin, J. H. 1989. Display based problem solving. In *Complex Information Processing: The Impact of Herbert A. Simon*, eds. D. Klahr and K. Kotovsky. Hillsdale, NJ: Erlbaum.

Laughery, K. R. 1989. Micro SAINT—A tool for modeling human performance in systems. In *Applications of Human Performance Models to System Design*, eds. G. R. McMillan, D. Beevis, E. Salas, M. H. Strub, R. Sutton, and L. Van Breda, 219–30. New York: Plenum Press. See also the web site of Micro Analysis and Design, Inc., http://www.maad.com.

McMillan, G. R., D. Beevis, E. Salas, M. H. Strub, R. Sutton, and L. Van Breda. 1989. *Applications of Human Performance Models to System Design*. New York: Plenum Press.

Merritt, F. S., M. K. Loftin, and J. T. Ricketts. 1996. *Standard Handbook for Civil Engineers.* New York: McGraw-Hill.

Newell, A. *Unified Theories of Cognition.* 1990. Cambridge, MA: Harvard University Press.

Norman, D. A. 1986. Cognitive Engineering. In *User Centered System Design*, eds. D. A. Norman and S. W. Draper. Hillsdale, NJ: Lawrence Erlbaum Associates.

Petrosky, H. 1985. *To Engineer is Human: The Role of Failure in Successful Design.* New York: St. Martin's Press.

Pew, R. W., S. Baron, C. E. Feehrer, and D. C. Miller. 1977. *Critical Review and Analysis of Performance Models Applicable to Man-Machine Systems Operation*. Technical Report No. 3446, Cambridge, MA: Bolt, Beranek and Newman, Inc.

Polson, P. G. 1987. A quantitative model of human-computer interaction. In *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*, ed. J. M. Carroll. Cambridge, MA: Bradford, MIT Press.

Pugsley, A. ed. 1976. *The Works of Isambard Kingdom Brunel: An Engineering Appreciation*. London and Bristol: The Institution of Civil Engineers and The University of Bristol.

St. Amant, R., T. E. Horton, and F. E. Ritter. 2007. Model-based evaluation of expert cell phone menu interaction. *ACM Trans Hum Comput Interact* 14(1):1–24.

St. Amant, R., and M. O. Riedl. 2001. A perception/action substrate for cognitive modeling in HCI. *Int J Hum Comput Stud* 55(1):15–39.

Strieb, M. I., and R. J. Wherry. 1979. *An Introduction to the Human Operator Simulator*. Technical Report 1400.02-D, Willow Grove, PA: Analytics Inc.

Vera, A. H., A. Howes, M. McCurdy, and R. L. Lewis. 2004. A constraint satisfaction approach to predicting skilled interactive cognition. In *Proceedings of the CHI 2004*, 121–8. New York: ACM.

Wherry, R. J. 1976. The human operator simulator—HOS. In *Monitoring Behavior and Supervisory Control*, eds. T. B. Sheridan and G. Johannsen. New York: Plenum Press.

Whiteside, J., S. Jones, P. S. Levy, and D. Wixon. 1985. User performance with command, menu, and iconic interfaces. In *Proceedings of CHI '85.* New York: ACM.

Wood, S. D. 2000. *Extending GOMS to Human Error and Applying it to Error-Tolerant Design*. Doctoral dissertation. Ann Arbor, MI: University of Michigan.

Zachary, W., T. Santarelli, J. Ryder, and J. Stokes. 2000. *Developing a Multi-Tasking Cognitive Agent Using the COGNET/IGEN Integrative Architecture*. Technical Report No. 001004.9915, Lower Gwynedd, PA: CHI Systems, Inc. See also the web site for CHI Systems, Inc., http://www.chiinc.com/.

58 Spreadsheet Tool for Simple Cost-Benefit Analyses of User Experience Engineering

Deborah J. Mayhew

CONTENTS

58.1 INTRODUCTION

This chapter offers and explains a free spreadsheet-based tool developed to help estimate the *potential return on investment (ROI)* of an investment in *user experience (UX) resources and activities* on software or website development projects. The spreadsheet-based tool is provided as a free download on my website (Mayhew 2009). The tool is a multi-worksheet Microsoft Excel file in .xls format. Table and page numbers referred to in this chapter as well as in the tool worksheets are references to table and page numbers in the book *Cost-Justifying Usability—An Update for the Internet Age* (Bias and Mayhew 2005)—a reference for much more detail on cost justifying UX efforts.

While the tool is general purpose and offers assistance for cost justifying UX resources in many different development contexts (e.g., internal desktop applications, commercial software, websites of various kinds), in this chapter, I explain the tool and how to use it in the context of cost justifying UX efforts when developing websites in particular.

Any cost-benefit analysis must start with estimating the *cost* of an investment, followed by predicting the potential *benefits*, so as to determine whether, and to what extent, the benefits might be expected to exceed the costs, that is, the extent to which an ROI will be realized. To estimate the costs of a UX engineering investment, one must have a plan for UX resources and activities as a part of the overall software or website design and development plan. My book *The Usability*

Engineering Lifecycle (Mayhew 1999) describes and explains a generic approach to usability engineering, which has since come to be more commonly referred to as UX engineering. It is this lifecycle that is the basis of estimating the cost side in the cost justification tool described and explained in this chapter. The usability engineering lifecycle is a general framework that is relevant to cost justifying a UX effort when developing any kind of software application, from commodity trading to an eCommerce website. The next section provides an overview of the usability engineering lifecycle, although wording has been slightly changed to reflect the use of the broader term "user experience" and the context of website development.

58.2 USER EXPERIENCE ENGINEERING LIFECYCLE—AN OVERVIEW

The first step in cost justifying a UX engineering effort on a website development project is to lay out a UX engineering plan for that project. This section provides a very high level synopsis of such a plan, based on *The Usability Engineering Lifecycle* (Mayhew 1999).

The UX engineering lifecycle consists of a set of UX tasks applied in a particular order at specific points in an overall website development lifecycle.

Several types of tasks are included in the UX engineering lifecycle:

- Structured UX requirements analysis tasks
- An explicit UX goal-setting task, driven directly from requirements analysis data
- Tasks supporting a structured, top-down approach to UX design that is driven directly from UX goals and other requirements data
- Objective UX evaluation tasks for iterating design toward UX goals

Figure 58.1 represents in summary, visual form, the usability (or UX) engineering lifecycle. The lifecycle is cast in three phases: requirements analysis, design/testing/development, and installation. Specific UX tasks within each phase are presented in boxes, and arrows show the basic order in which tasks should be carried out. Much of the sequencing of tasks is iterative, and the specific places where iterations would most typically occur are illustrated by arrows returning to earlier points in the lifecycle. Brief descriptions of each lifecycle task follow, again with slight wording changes relative to the chart in Figure 58.1 to reflect website development in particular.

58.2.1 Phase 1: Requirements Analysis

58.2.1.1 User Profile

A description of specific user characteristics relevant to UX design (e.g., level of general computer and/or web literacy, expected frequency of use, level of familiarity with tasks supported by the website) is obtained for the intended user population. This will drive tailored UX design decisions and also identify major user categories for study in the Task Analysis task.

58.2.1.2 Task Analysis

A study of users' natural task flow patterns, informational needs, and conceptual frameworks is conducted, resulting in an understanding and specification of underlying user goals. These will be used to set UX goals and drive information architecture design (previously referred to as work reengineering) and UX design.

58.2.1.3 Platform Capabilities/Constraints

The UX design capabilities and constraints inherent in the technology platform chosen for the product (e.g., browsers and browser versions, screen sizes and resolutions) are determined and documented. These will define the scope of possibilities for UX design.

58.2.1.4 General Design Guidelines

Relevant general UX design guidelines available in the UX engineering literature are gathered and reviewed. They will be applied during the design process to come, along with all other project-specific information gathered in the previous tasks.

58.2.1.5 User Experience Goals

Specific *qualitative* goals reflecting UX requirements are developed, extracted from the user profile and task analysis work products. In addition, *quantitative* goals (based on a subset of high-priority qualitative goals) may be developed, defining minimal acceptable user performance and satisfaction criteria. These UX goals focus later design efforts and form the basis for later iterative UX evaluation.

58.2.2 Phase 2: Design/Testing/Development

58.2.2.1 Level 1 Design

58.2.2.1.1 Information Architecture

In this task (previously referred to as work reengineering) based on all requirements analysis data and the UX goals extracted from them, user tasks are redesigned at the level of organization and workflow to streamline user tasks and exploit the capabilities of automation. No *visual* UX design is involved in this task, just abstract organization of functionality and workflow design. The information architecture defines how users will navigate through the information and/or functionality of the application.

58.2.2.1.2 Conceptual Model Design

Based on all the previous tasks, a set of design conventions are generated for visually presenting the different levels in the (usually hierarchical) information architecture and for interactions for navigating through it. Page design detail is *not* addressed at this design level.

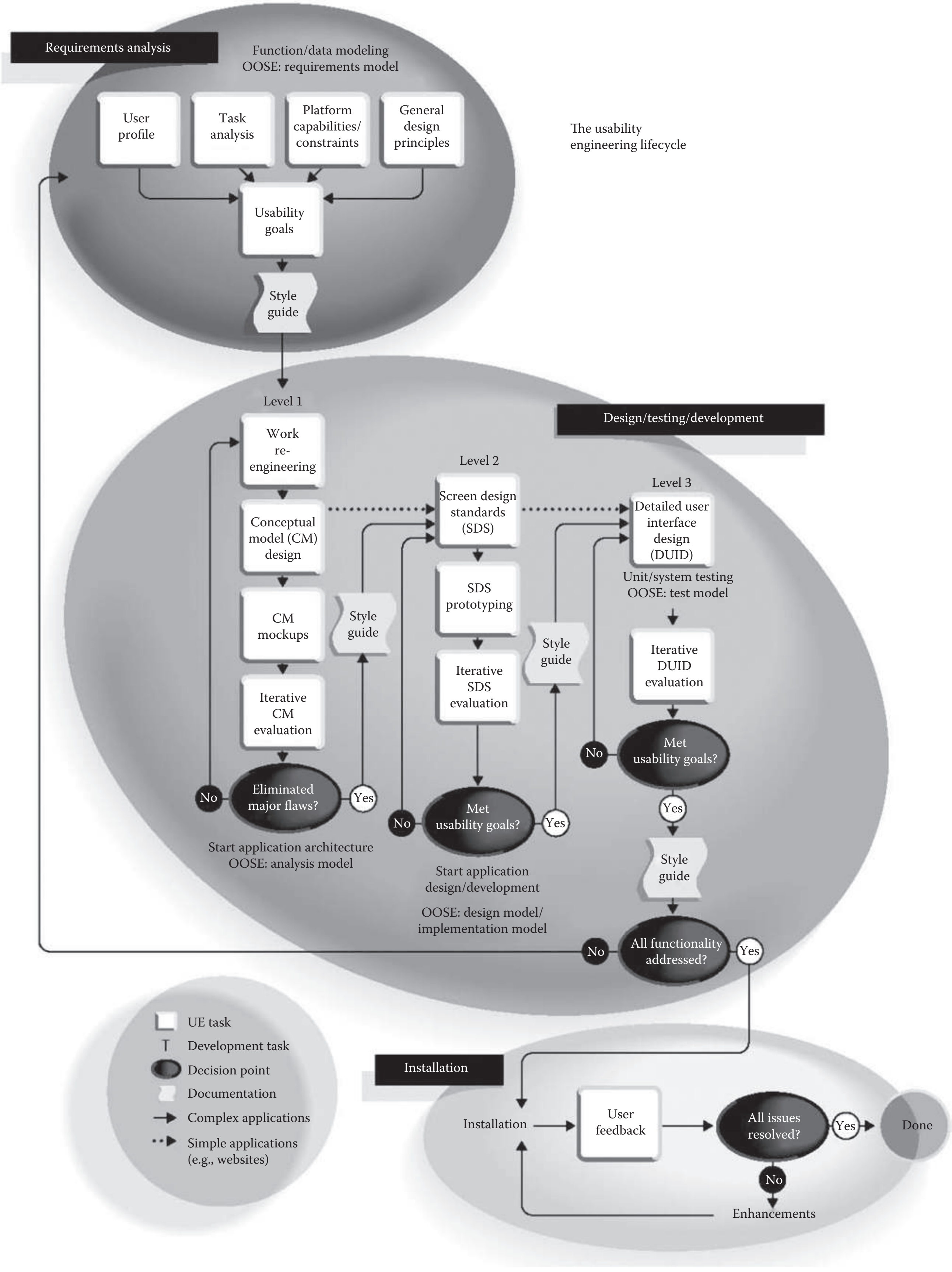


FIGURE 58.1 The usability engineering lifecycle. (From Mayhew, D. J. 1999. *The Usability Engineering Lifecycle*. San Francisco, CA: Morgan Kaufmann Publishers.)

58.2.2.1.3 Conceptual Model Mockups

Paper-and-pencil or live prototype mockups of high-level design ideas generated in the conceptual model design task are prepared, representing ideas about high-level functional organization and conceptual model design. Detailed page design and complete functional design are *not* in focus here.

58.2.2.1.4 Iterative Conceptual Model Evaluation

The mockups are evaluated and modified through iterative evaluation techniques such as formal UX testing, in which real, representative end users attempt to perform real, representative tasks with minimal training and intervention, imagining that the mockups are a real website UX design. This and the previous two tasks are conducted in iterative cycles until all major UX "bugs" are identified and engineered out of Level 1 (i.e., Conceptual Model) Design. Once a conceptual model is relatively stable, system architecture design can commence.

58.2.2.2 Level 2 Design

58.2.2.2.1 Page Design Standards

A set of website-specific standards and conventions for all aspects of detailed page design is developed, based on any corporate standards that have been mandated, the data generated in the requirements analysis phase, and the product-unique conceptual model design arrived at during Level 1 Design. Page design standards will ensure another level of coherence and consistency—the foundations of usability—across the website UX.

58.2.2.2.2 Page Design Standards Prototyping

The page design standards (as well as the conceptual model design) are applied to design the detailed UX to selected subsets of website functionality. This design is implemented as a running prototype.

58.2.2.2.3 Page Design Standards Evaluation

An evaluation technique such as formal UX testing is carried out on the page design standards prototype and then redesign/re-evaluation iterations are performed to refine and validate a robust set of page design standards. Iterations are continued until all major UX "bugs" are eliminated and UX goals seem within reach.

58.2.2.2.4 Style Guide

At the end of the design/evaluate iterations in design levels 1 and 2, you have a validated and stabilized conceptual model design and a validated and stabilized set of standards and conventions for all aspects of detailed page design. These are captured in a document called a style guide (and ultimately in the CSS style sheets in the website code), which already documents the results of requirements analysis tasks. During detailed UX design, which comes next, following the conceptual model design and page design standards in the website style guide will ensure quality, coherence, and consistency—the foundations of a good UX design.

58.2.2.3 Level 3 Design

58.2.2.3.1 Detailed UX Design

Detailed design of the complete website UX is carried out based on the refined and validated information architecture, conceptual model design, and page design standards documented in the website style guide. This design then drives website development.

58.2.2.3.2 Iterative Detailed User Experience Design Evaluation

An evaluation technique such as formal UX testing is continued during website development to expand evaluation to previously unassessed subsets of functionality and categories of users and also to continue to refine the UX and validate it against UX goals.

58.2.3 Phase 3: Installation

58.2.3.1 User Feedback

After the website has been launched and in production for some time, feedback is gathered to feed into design enhancements, design of new pages, and/or design of new but related websites.

58.3 GENERAL APPROACH TO COST-BENEFIT ANALYSIS OF USER EXPERIENCE ENGINEERING

To cost justify a proposed UX engineering plan, you simply adapt a very generic and widely used cost-benefit analysis technique. Having laid out a detailed UX engineering project plan based on The Usability Engineering Lifecycle (see above, and Mayhew 1999), it is a fairly straightforward matter to calculate the costs of that plan. Then you need to calculate the predicted benefits. This is a little trickier, and it is where the adaptation of the generic cost justification analysis comes into play. Then, you simply compare costs to benefits to find out if, when and to what extent, the benefits are predicted to outweigh the costs. If they do to a satisfactory extent, then you have cost justified the planned UX engineering effort. If they do not, then you will need to rethink your UX engineering plan.

More specifically, first a UX engineering plan is laid out. The plan specifies particular techniques to use for each UX engineering task in the lifecycle, breaks the techniques down into steps, and specifies the personnel hours and equipment costs for each step. The *cost* of each task is then calculated by multiplying the total number of hours for each type of personnel (e.g., UX professionals, developers, test participants) by their effective hourly wage (fully loaded, i.e., including salary, benefits, office space, equipment, utilities, and other facilities) and adding up personnel costs across types. Sometimes it is hard to get data on fully loaded wages for an organization. In this case, I use a rule of thumb I have heard informally to simply double the before-tax annual salary and then divide by the typical number of hours a full time worker is paid for in a year, usually about 2000. Even if my audience is unwilling or unable to give me actual figures for fully loaded wages, they can contest—or not—my ballpark figure based on this rule of thumb. Then the costs from all tasks are summed to arrive at a total estimated cost for the plan.

Next, the overall *benefits* of the specific UX engineering plan are predicted by selecting relevant benefit categories, calculating expected benefits by plugging project-specific parameters and assumptions into benefit formulas, and summing benefits across categories.

The potential benefit categories relevant to a particular cost-benefit analysis will depend on the basic business model of the software being developed. Benefit categories potentially relevant to different types of software applications and websites are summarized in Figure 58.2 and in the Benefits Categories worksheet of the spreadsheet tool (see also Table 3.1 on page 58 of Bias and Mayhew 2005). This would be a good time to download and open the spreadsheet tool (Mayhew 2011).

Note that the relevant benefit categories for different types of applications/websites vary somewhat. In a cost-benefit analysis, one wants to focus attention on the potential benefits that are *of most relevance to the bottom line business goals for an application or website*, either short term or long term or both.

Note also that these benefits represent just a sample of those that might be relevant for the types of software applications and websites listed. Other benefits relevant to these particular types of projects might be included as appropriate, given the business goals of the application or website stakeholders and the primary concerns of the audience, and could be calculated in a similar fashion within the spreadsheet tool as those described later. And of course, very different kinds of automation projects exist that may have very different kinds of expected benefits, for example, lives and equipment saved and wars won in a military context. The general cost justification approach can be applied in these latter types of

	Application/Site Type							
Benefits	**Internal Application**	**Commercial Product**	**E-Commerce Site**	**E-Services Site**	**Site Funded by Advertising**	**Product Information Site**	**Customer Service Site**	**Intranets**
Increased buy-to-look ratios			✓					
Decreased abandoned shopping carts			✓					
Increased number of visits					✓			
Increased return visits			✓		✓			
Increased length of visits					✓			
Decreased failed searches			✓		✓			
Decreased costs of other sales channels			✓					
Decreased use of “call back” button (i.e., live customer service)			✓	✓			✓	✓
Savings due to making changes earlier in development lifecycle	✓	✓	✓	✓	✓	✓	✓	✓
Increased “click though” on ads					✓			
Increased sales leads						✓		
Increased sales		✓						
Decreased costs of traditional customer service channels		✓		✓			✓	
Decreased training costs	✓	✓						✓
Increased user productivity	✓							✓
Decreased user errors	✓							✓

FIGURE 58.2 Benefit categories by application/site type.

situations, but the spreadsheet tool would have to be significantly modified to address them.

Finally, overall predicted benefits are compared to overall estimated costs to see if, and to what extent, the overall UX engineering plan is justified as an investment.

When UX practitioners are invited to participate in application or website development projects already in progress, which is often the case, it may be difficult to include all tasks and to influence overall schedules and budgets. They are likely to have to live within already-committed-to schedules, assumed platforms and system architectures, use shortcut techniques for lifecycle tasks, and minimally impact budgets. Nevertheless, it is almost always possible to create a UX engineering plan that will make a significant contribution to an application or website development project, even when one comes in relatively late. And, you can use the cost-benefit analysis technique to prepare and support even plans that involve only parts of the overall lifecycle and only shortcut techniques for tasks within it.

58.4 EXAMPLE OF COST-BENEFIT ANALYSIS

In this section, I provide both a concrete example of conducting a cost-benefit analysis of a UX engineering plan for a website and introduce and explain the use of my spreadsheet-based cost justification tool, available as a free download from my website (Mayhew 2009). The example I use involves the development of an eCommerce website.

58.4.1 An eCommerce Website Development Project

First, let us look at the overall results of the cost-benefit analysis for this example, both in the spreadsheet tool and the book if you have it (Bias and Mayhew 2005). We will assume the UX project plan with its associated cost that is shown in Figure 58.3 and presented in the Total Costs worksheet of the spreadsheet tool (see also Table 3.2 on page 61 of Bias and Mayhew 2005).

In this example, the project UX engineer estimated that in the case of this project, the UX engineering plan shown in Figure 58.3 would produce an eCommerce website with the *expected benefits* summarized in Figure 58.4 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.26 on page 87 of Bias and Mayhew 2005).

Comparing these benefits and costs, the UX engineer argued that the proposed UX engineering plan would more than pay for itself in the first year after launch, as shown in Figure 58.5 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.27 on page 87 of Bias and Mayhew 2005).

▨ Your data ░ Formulas

Cost Calculations						
Phase	Task (Technique)	Usability Engineers Hours @	Developers Hours @	Managers Hours @	Users Hours @	Total Cost
		$175	$175	$200	$25	
Requirements Analysis	User profile (questionnaire)	62	0	4	33	$12,475
	Contextual task analysis	138	8	8	80	$28,650
	Platform capabilities and constraints	16	6	0	0	$3,850
	Usability goals	20	0	4	2	$4,350
Design/Testing/Development	Work reengineering (information architecture)	80	0	0	16	$14,400
	Conceptual model design	80	8	0	8	$15,600
	Conceptual model mockups (paper prototype)	36	0	0	0	$6,300
	Iterative conceptual model evaluation (usability test)	142	0	0	22	$25,400
	Screen design standards	80	8	0	8	$15,600
	Screen design standards prototyping (live prototype)	28	80	0	0	$18,900
	Iterative screen design standards evaluation (usability test)	142	40	0	22	$32,400
	Detailed user interface design	80	8	0	8	$15,600
	Iterative detailed user interface design evaluation (usability test)	142	40	0	22	$32,400
	Totals	1046	198	16	201	$225,926

FIGURE 58.3 Cost of a user experience engineering plan.

Benefit Category	Benefit Value—*per Month*
Increased buy-to-look ratio:	$12,500.00
Decreased abandoned shopping carts:	$12,500.00
Decreased usage of "call back" button:	$3,125.00
Total monthly benefit:	$28,125.00

FIGURE 58.4 Expected benefits per month for an eCommerce website.

Net Benefit Calculations	
Benefits per month:	$28,125.00
Total cost:	$225,925.00
Pay off period in months (cost/benefit):	8.03
Net benefit in first year:	$111,575.00

FIGURE 58.5 Net benefit calculations for an eCommerce website.

Analysis Parameters	Values
Application type:	eCommerce Website
Current average visitors per month:	125,000
Current buy-to-look ratio:	2%
Current rate of usage of the "call back" button:	2%
Profit margin per unit:	$10
Average length of servicing each use of "call back" button (3 minutes expressed as hours):	0.050000
User fully loaded hourly wage:	$25
Developer fully loaded hourly wage:	$175
Usability engineer fully loaded hourly wage:	$175
Manager fully loaded hourly wage:	$200
Customer support fully loaded hourly wage:	$50
Usability lab:	In place

FIGURE 58.6 Analysis parameters for an eCommerce website.

Note that the simple analyses offered in this example and in the spreadsheet-based tool do not consider the time value of money, that is, the money for the costs is spent at one point in time, whereas the benefits come later in time, and this is *not* taken into account in this simple analysis. Also, if the money was *not* spent on these costs, but instead was invested in some other way, this money would likely increase in value and this is not addressed either. In my experience, usually the predicted benefits of UX engineering are so dramatic that these more sophisticated financial considerations are not necessary to convince the audience for the analysis. However, if needed, these more sophisticated calculations based on the time value of money and other factors are explained in Chapter 7 of Bias and Mayhew (2005).

The project UX engineer expected the UX engineering plan in this example to be approved on the basis of the cost justification in this example. Now let us see exactly how this net benefit was calculated, using the spreadsheet tool.

58.4.1.1 Start with the User Experience Engineering Plan

This is the first step in conducting a cost-benefit analysis. The UX engineering plan identifies which UX engineering lifecycle tasks and techniques (see above and also Mayhew 1999) will be used and breaks them down into required staff and hours. Costs can then be computed for these tasks in the next two steps below.

In this example, we start with the assumed plan represented in Figure 58.3 and in the Total Costs worksheet of the spreadsheet tool (see also Table 3.2 on page 61 of Bias and Mayhew 2005). Note that almost all the cells in this worksheet have a *pink* (gray dotted in the illustrations in this chapter) background. This indicates that the values in these cells are computed according to *formulas* that reference cells in other worksheets. Thus, when using the tool, you should *not* directly edit any of the pink (gray dotted) cells in the worksheets. Cells you *should* edit to reflect your particular project will appear with a *green* background in worksheets (solid gray in the illustrations in this chapter). For this stage in the analysis process, we are really just focused on the first two columns of this table, which lay out the UX engineering plan.

It is important to note that there is never one correct UX plan. This, as much else, is something that will vary across projects. The choice of technique for carrying out each task in the UX engineering lifecycle will depend on project budget and schedules as well as application/website complexity. Thus the example plan presented here should not be assumed. A project-unique plan must be designed around the parameters of a specific project.

58.4.1.2 Establish Analysis Parameters

Most of the calculations for both estimated costs and predicted benefits are based on project-specific parameters. These should be researched, established, and documented before proceeding with the analysis. Analysis parameters for this example are presented in Figure 58.6 and in the eCommerce worksheet of the spreadsheet tool (see also Table 3.28 on page 88 of Bias and Mayhew 2005).

In this example, we assume there is an existing website with known parameters, and the project involves a redesign, a common scenario for web development today.

It should be emphasized that when using the general cost-benefit analysis technique illustrated here, the particular parameters and parameter *values* used in this example should *not* be assumed. Both the particular parameters themselves and the parameter values of *your* project and organization should be substituted for those in Figure 58.6 and in the eCommerce worksheet (as well as all other worksheets) of the spreadsheet tool (see also Table 3.28 on page 88 of Bias and Mayhew 2005). For example, your current website traffic will be totally different, and the fully loaded hourly wage (the costs of salary plus benefits, office space, equipment, utilities, and other facilities) of your personnel may be significantly lower or higher than those assumed in these sample analyses. See also other worksheets in the spreadsheet tool for examples of relevant parameters for other types of development projects.

Note that in general, certain parameters in a cost-benefit analysis have a major impact on the magnitude of potential benefits. For example, when considering eCommerce websites as in this example, the critical parameters are the current average visitors per month and the average profit margin of a sale. When there is a high volume of site traffic and the average profit margin of a sale is high, even very small increases in conversion rates in an optimized website UX will add up quickly to significant overall benefits. On the other hand, where there is light traffic and/or a low profit margin per sale, benefits may not add up to anything very significant.

For example, consider the following two scenarios. First, imagine a case where average visitors per month are 10,000 and the average profit margin of a sale is $100. Even a conversion rate 1% higher than the current rate in this case adds up:

$$\begin{aligned}&10{,}000 \text{ visitors per month} * \text{conversion rate increase of } 1\% *\\ &\quad \$100 \text{ per sale } = \$10{,}000 \text{ more in profit, per month}\end{aligned}$$

This is a pretty dramatic benefit for a small improvement on the conversion rate.

On the other hand, if there were only 1000 visitors per month and the average profit margin per sale was $10, even if a 1% increase in conversion rate could be realized, the overall benefit would only be

$$\begin{aligned}&1{,}000 \text{ visitors per month} * \text{conversion rate increase of } 1\% *\\ &\quad \$10 \text{ per sale } = \$100 \text{ more profit, per month}\end{aligned}$$

Thus, in the case of eCommerce websites, costs associated with optimizing the UX are more likely to pay off when both traffic and the average profit margin per sale are high.

58.4.1.3 Calculate the Cost of Each User Experience Engineering Lifecycle Task in the User Experience Engineering Plan

The cost of each individual task/technique listed in Figure 58.3 and in the Total Costs worksheet of the spreadsheet tool (see also Table 3.2 on page 61 of Bias and Mayhew 2005) was estimated by breaking the task/technique down into small steps, estimating the number of hours required for each step by different types of personnel and multiplying these hours by the known fully loaded hourly wage of each type of personnel (if outside consultants or contractors are used, their simple hourly rate plus travel expenses would apply, and if external users are recruited to participate, they will be paid at some simple hourly rate or flat fee).

In our example, the project UX engineer used the task cost calculations, as shown in the User Profile through DUID worksheets of the spreadsheet tool. One example of these task cost calculations can be seen in Figure 58.7 (see also Tables 3.6 through 3.18 on pages 66–72 in Bias and Mayhew 2005).

These task cost calculations show the derivation of the numbers summarized in Figure 58.3 and in the Total Costs worksheet of the spreadsheet tool (see also Table 3.2 on page 61 of Bias and Mayhew 2005).

User Profile (Questionnaire)	Usability Engineers	Developers	Managers	Users	
Step	**Hours**	**Hours**	**Hours**	**Hours**	
Needs finding	4		2	2	
Draft questionnaire	6				
Management feedback	2		2	2	
Revise questionnaire	4				
Pilot questionnaire	4			4	
Revise questionnaire	2				
Select user sample	4				
Distribute questionnaire/ respond	8			25	
Data analysis	8				
Data interpretation/presentation	20				
Total hours	62	0	4	33	
Times hourly rate	$175	$175	$200	$25	
Equals	$10,850 plus	$0 plus	$800 plus	$825	= $12,475

FIGURE 58.7 Cost of user profile (Questionnaire).

Developer	$175
Usability engineer (UE)	$175
User	$25
Manager	$200
Customer support	$50
Trainer	$50

FIGURE 58.8 Fully loaded hourly rates.

One of the parameters used in the calculations of cost is fully loaded hourly wage of involved personnel. Fully loaded hourly wages are calculated by adding together the cost of salary, benefits, office space, equipment, and any other relevant overhead for a type of personnel and dividing this by the number of hours paid for each year by that personnel type. The hourly rate used here for UX staff is based on an informal average of typical current salaries of senior level internal UX staff and external consultants (see UPA website 2009 for the most recent salary survey of usability practitioners). The hourly rate of developers was similarly estimated (see, e.g., Payscale website 2011). However, the fully loaded hourly rate figures used to generate this and the other sample cost-benefit analyses in the spreadsheet tool and Bias and Mayhew (2005) are just examples, and you would have to substitute the actual hourly rates of personnel in your own organization in an actual analysis. Additional costs, such as equipment and supplies, could also be estimated and added into the total cost of each task/technique, although that was not done in this example for simplicity's sake.

In the spreadsheet tool, you would start by plugging in your fully loaded hourly wage parameters in the Fully Loaded Hourly Rate worksheet, as shown in Figure 58.8.

Then, for each table in the User Profile through the DUID Evaluation worksheets, you would enter your planned level of effort for each task, as for example in Figure 58.7 (see also Tables 3.6 through 3.18 on pages 66–72 in Bias and Mayhew 2005). *For any task that is not included in your plan, it is important to enter zeroes in all gray (green) cells* in the worksheet for that task. Similarly, if you plan to conduct a given task but not every listed step in it, simply enter zeroes in the gray (green) cells of any steps you do not plan to conduct.

Once you have entered your unique project parameters in the hourly wage and tasks worksheets, you can look back at the Total Costs worksheet to see the total estimated cost of your plan.

58.4.1.4 Select Relevant Benefit Categories

Since our example is a redesign of an *eCommerce website*, only certain benefit categories are of relevance to the business goals of the project, as shown in Figure 58.2 and the Benefits Categories worksheet in the spreadsheet tool (see also Table 3.1 on page 58 in Bias and Mayhew 2005). These include the following:

- Increased buy-to-look ratio
- Decreased abandoned shopping carts
- Decreased usage of "call back" button

As discussed earlier, the best benefit categories to include in a cost-benefit analysis will depend on the type of project and the intended audience for the analysis.

In this example, as compared to the existing site design, the UX engineer anticipated that in the course of redesign, the UX engineering effort would decrease abandoned shopping carts by insuring that the checkout process is clear, efficient, provides all the right information at the right time, and does not bother users with tedious data entry of information they do not want or need to provide. She/he expected to improve the buy-to-look ratio by insuring that the right product information is contained on the site and that navigation to find products is efficient and always successful. She/he expected to decrease the use of the "call back" button by insuring that the information architecture matched users' expectations and by designing and validating a clear conceptual model, so that navigation of and interactions with the site are intuitively obvious. Accomplishing all these things depends upon conducting the requirements analysis and testing activities in the proposed plan, as well as on applying general UX design expertise.

Figure 58.4 and the eCommerce worksheet in the spreadsheet tool (see also Table 3.26 on page 87 in Bias and Mayhew 2005) summarize the predicted magnitude of each of these benefit categories and then sums across them to predict a total benefit.

When selecting benefit categories to use in your own cost justification analysis of a UX engineering plan for an eCommerce website development project, you can use the benefits assumptions table in the eCommerce worksheet in the spreadsheet tool (also shown in Figure 58.9) to make your benefit assumptions. You can use all three of the benefit categories in that table, or choose to use any combination of them, or even others that are relevant to your situation. Simply enter zeroes in the cells representing assumption values for benefit categories you do not wish to include.

What follows is an explanation of how benefit predictions in each category were derived from project-specific parameters and assumptions.

58.4.1.5 Predict Benefits

In this step, the project UX engineer predicted the magnitude of the benefits that would be realized—*relative to the current website,* which is being redesigned—*if* the UX engineering plan (with its associated costs) was implemented. Benefits were predicted in each selected benefit category by doing some simple arithmetic based on project-specific analysis parameters and some simple project-specific assumptions.

Note that while at this point in the process of your own analysis, you have already filled in your project-unique parameters, you now need to consider your project-unique benefits assumptions and modify the values for them in the benefits assumptions table in the spreadsheet tool. The project *parameters* for this example are laid out in Figure 58.6 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.28 on page 88 in Bias and Mayhew 2005). The benefits *assumptions* are given in Figure 58.9 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.29 on page 90 in Bias and Mayhew 2005).

In the case of calculating predicted buy-to-look ratio benefits, the relevant *parameters* are from Figure 58.6 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.28 on page 88 in Bias and Mayhew 2005):

- Current average visitors per month
- Profit margin per unit

The *assumption* made regarding increased buy-to-look ratio in our example (see Figure 58.9 and the eCommerce worksheet in the spreadsheet tool) is the following:

- There will be a 1% increase in visitors who decide to buy (and check out successfully)

Benefit assumptions are the crux of the whole cost-benefit analysis. While *costs* can be calculated with a high degree of confidence based on past experience, and all the *parameters* fed into the analysis are known facts, the *assumptions* made are just that—assumptions, predictions, rather than known facts or guaranteed outcomes. The audience for the analysis is asked to accept that these assumptions are reasonable ones and they must to be convinced by the overall analysis.

It should be pointed out that *any* cost-benefit analysis for *any* purpose must ultimately include some assumptions that are really only predictions of the likely outcome of investments of various sorts. The whole point of a cost-benefit analysis is to try to evaluate in advance, in a situation in which there is some element of uncertainty, the likelihood that an investment will pay off. The trick is basing the prediction of ROI on a firm foundation of known facts and reasonable assumptions. In the case of a cost-benefit analysis of a UX engineering effort, there are several foundations upon which to formulate sound assumptions regarding predicted benefits, including the following:

- References to the general UX literature documenting impacts of certain types of design approaches on user performance
- References to after-the-fact case studies of benefits achieved through UX engineering
- Anecdotes from colleagues
- One's own past experiences as a UX engineer
- One's own past experience working with a particular design/development organization

See Bias and Mayhew (2005) for more in-depth discussion of making and supporting analysis assumptions.

In general, it is usually wise to make very conservative benefit assumptions, for several reasons. First, any cost-benefit analysis has an intended audience, who must be convinced that benefits will in fact outweigh costs. Assumptions that are very conservative are less likely to be challenged by the relevant audience, thus increasing the likelihood of acceptance of the analysis conclusions. In addition, conservative benefit assumptions help to manage expectations. It is always better to achieve a greater benefit than was predicted in the cost-benefit analysis, than to achieve a lesser benefit, even if it still outweighs the costs. Having underestimated benefits will likely make future cost-benefit analyses more credible and more readily accepted. Also, it is important to realize that some validly predicted benefits may be cancelled out by other non-UX-related changes, such as decreases in user morale and motivation, decreased system reliability or response time, and so on. Having made conservative benefit predictions decreases the possibility that other factors will completely wipe out any benefits due to improved usability.

Returning to the explanation of the derivation of benefit predictions in our example, we see the benefit assumptions for each benefit category given in Figure 58.9 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.29 on page 90 in Bias and Mayhew 2005). The UX engineer based the benefit assumptions in this analysis on statistics available in the literature. In particular, she/he began with the often quoted average eCommerce website buy-to-look ratio of 2%–3% (Sonderegger 1998; Souza 2000, Fireclick 2010). She/he then based the assumption that she/he could improve this rate by a minimum of 2% (1% from improving the product search process and 1% from improving the checkout process) through UX engineering techniques, on a variety of statistics available in the literature.

Benefit Assumptions	
Increased Buy-to-Look Ratio	
Increase in visitors who decide to buy (and checkout successfully), relative to total monthly visitors	1%
Decreased Abandoned Shopping Carts	
Increase in visitors who have already decided to buy but who also now checkout successfully, relative to total monthly visitors	1%
Decreased Usage of "call back" button	
Decrease in number of visitors using "call back" button, relative to total monthly visitors	1%

FIGURE 58.9 Benefits assumptions for an eCommerce website.

For example, Souza (2001) suggests that it is typical for as much as 5% of online shoppers to fail to find the product and offer they are looking for and cites one study in which 65% of shopping attempts at a set of prominent eCommerce sites ended in failure. Sonderegger (1998) suggests sales under perform on eCommerce sites by as much as 50% or more due to poor site usability. GVU (1999) survey data suggests that almost 50% of website users cannot find the information they are looking for and that over 80% web shoppers have left one site for another when they had dissatisfying experiences with site usability. Souza (2001) also notes that companies spend between $100,000 and $1,000,000 on site redesigns, but few have any sense of which specific design changes might pay off.

The project UX engineer based the assumption of reduced usage of the "call back" button by 1% on statistics, suggesting that as much as 20% of eCommerce site users typically call in to get more information (Souza 2001).

Most of us have experienced all these problems—difficulty finding products, difficulty checking out, and need to use a "call back" button to complete transactions—and would have little argument with the idea that they are typical even in 2012. Given all the dramatic statistics cited, the modest assumptions made in this analysis seem very conservative indeed. And given the fact that companies typically spend a great deal of money on redesign with no process in place that can insure improvements in the site UX, the notion of a highly structured and goal-oriented UX process starts to make a lot of sense.

Figure 58.10 and the eCommerce worksheet in the spreadsheet tool (see also Table 3.30 on page 91 in Bias and Mayhew 2005) show the calculation of the total predicted benefit in each benefit category, based on parameters and assumptions in Figures 58.6 and 58.9.

The sum of predicted benefits in these three separate benefits categories is given in Figure 58.4—a monthly benefit of $28,125.

When conducting your own project-specific cost-benefit analysis, note that while you need to enter values for some parameters and for assumptions in the project type worksheet in the spreadsheet tool, all tables to the right of those two in that worksheet are based on calculations involving previously entered values in other tables and worksheets, indicated by the dotted gray (pink) background in those cells. You will not edit those tables at all. Also recall that you will want to enter zeroes for any parameter or assumption you do not wish to use in your analysis.

58.4.1.6 Compare Costs to Benefits

Having calculated the costs of a particular UX engineering plan and predicted the total benefits to result from executing that plan as compared to not executing it, the next step is simply to compare the total costs to the total benefits to arrive at a net benefit. In this example, this calculation is shown in Figure 58.5 and in the eCommerce worksheet in the spreadsheet tool (see also Table 3.27 on page 87 in Bias and Mayhew 2005). The analysis predicts a clear and substantial net benefit ($111,575) in the first year. After that, additional benefits of $28,125 per month are predicted to continue to accrue assuming all else (e.g., the economy) remains constant.

Our UX engineer's initial UX engineering plan appeared to be well justified. It was a fairly aggressive plan, in that it included all lifecycle tasks, and the most reliable and thorough techniques for each task. Given the very clear net benefit, the UX engineer would have been wise to stick with this aggressive plan and submit it to project management for approval and funding.

If the net benefit had been marginal or if there had in fact been a net cost, then it would have been well-advised to go back and rethink the proposed UX engineering plan, scaling back to shortcut techniques for some tasks. Perhaps, for example, the UX engineer should have planned to do only a shortcut user profile by interviewing sales and marketing personnel, a shortcut task analysis consisting of just a few rounds of contextual observations/interviews with users, and do just one iterative cycle of UX testing on a complete detailed design, to catch major flaws and be sure the predicted benefits have been achieved. Of course, this would make the predictions more risky and call for an even more conservative analysis.

Individual Benefit Calculations				
Increased Buy-to-Look Ratio				
Current monthly visitors ×	Rate of increase in buyers ×	Profit margin per unit =		
125,000	1%	$10		$12,500.00
Decreased Abandoned Shopping Carts				
Current monthly visitors ×	Rate of increase in buyers ×	Profit margin per unit =		
125,000	1%	$10		$12,500.00
Decreased Usage of "call back" button				
Current monthly visitors ×	Rate of decrease in use of "call back" button ×	Hours saved per call eliminated ×	Customer support hourly rate =	
125,000	1%	0.050000	$50	$3,125.00

FIGURE 58.10 Benefits calculations for an eCommerce website.

As explained earlier, to plan the budget for a UX engineering program, it makes sense to start out by calculating the costs of the most aggressive UX engineering plan that you would like to implement, including the more reliable and thorough techniques for most if not all lifecycle tasks. If predicted benefits outweigh costs dramatically, as they usually will when critical parameters are favorable, then you can easily make a good argument for even the most aggressive UX engineering plan, because only the most conservative claims concerning potential benefits have been made, and as such can be defended easily.

If, however, costs and benefits in the initial calculation seem to match up fairly closely, then you might want to consider scaling back the planned UX engineering plan, maybe even to just a bare-bones plan, with more shortcut techniques applied for each lifecycle task.

To illustrate this planning strategy, consider the following two scenarios. First, revisit our example analysis, which involved an eCommerce website. A fairly conservative assumption was made concerning benefits: a 1% increase in conversion rate. Even with this fairly conservative assumption, the fairly aggressive UX engineering plan was predicted to pay off in the first year, with net benefits continuing to accrue dramatically after that.

In fact, if you had made the more aggressive and yet still realistic benefits assumption of a 3% increase in conversion rate (rather than by 1%), the net benefits would have summed to $411,575 in the first year alone, with benefits of $53,125 per month continuing to accrue after that. Thus, one could argue that while even the most conservative assumptions predict a fairly dramatic payoff of a comprehensive UX engineering plan, the likelihood is that the payoff will be higher still. Please note that this calculation is performed for you by the spreadsheet tool by simply changing a single cell—changing the benefit assumption of increased buy-to-look ratio from 1% to 3%.

In contrast, suppose you again started out by costing out a comprehensive UX engineering plan at $225,925. In this case, however, suppose that there are only 25,000 average visitors per month rather than 125,000. In this case, calculations using the original more conservative benefits assumptions would show a loss until well into the fourth year (try changing this single number in the parameters table in the worksheet and then inspecting the bottom line results in the last table to the right in the worksheet).

Even though the benefits assumptions were conservative, while a loss in the first few years is not necessarily a bad thing, it still seems risky to make an aggressive investment that, based on conservative assumptions, really does not show a significant payoff even over the course of four years. In this case, one would want to scale back the planned usability engineering program and its associated costs. Because the benefits assumptions made were so conservative, it is likely that they will be achieved even with a minimal usability effort. In this way, you can use the spreadsheet-based cost-benefit analysis tool to “what if” to plan a level of UX engineering effort that is most likely to pay off—or to decide to spend your limited UX dollars on a different project altogether that is more likely to pay off.

58.5 SUMMARY

The cost-benefit analysis example offered in this chapter is based on a simple subset of all actual costs and potential benefits and very simple and basic assumptions regarding the value of money over time. More complex and sophisticated analyses can be calculated (see Karat, Chapter 4 in Bias and Mayhew 2005). However, often a simple and straightforward analysis of the type offered in the example above will be sufficient for the purpose of winning funding for UX engineering investments during software application or website development in general or for planning appropriate UX engineering plans for specific development projects.

The example analysis offered here suggests that it is usually fairly easy to justify a significant investment of time and money in UX engineering during the development of eCommerce websites. The framework and example presented in this chapter, along with the free spreadsheet tool available from my website, should help you demonstrate that this is the case for your development projects.

REFERENCES

Bias, R. G., and D. J. Mayhew. 2005. *Cost Justifying Usability—An Update for the Internet Age*. San Francisco, CA: Morgan Kaufmann Publishers.

Fireclick. 2010. http://index.fireclick.com/. Last accessed 2010.

GVU. 1999. http://www.gvu.gatech.edu/user_surveys. Atlanta, GA: Georgia Institute of Technology. (No longer online, last accessed 2005.)

Mayhew, D. J. 1999. The Usability Engineering Lifecycle. San Francisco, CA: Morgan Kaufmann Publishers.

Mayhew, D. J. 2011. Deborah J. Mayhew & Associates website, Downloads page. http://drdeb.vineyard.net/index.php?loc=12&nloc=1. Last accessed November 28, 2011.

Payscale website. 2011. Career planning page. http://www.payscale.com/research/US/Country=United_States/Salary. Last accessed November 28, 2011.

Sonderegger, P. 1998. The Age of Net Pragmatism. Cambridge, MA: Forrester Research.

Souza, R. K. 2000. The Best of Retail Site Design. Cambridge, MA: Forrester Research.

Souza, R. K. 2001. Get ROI From Design. Cambridge, MA: Forrester Research.

UPA website. 2009. Resources, salary surveys page. http://usabilityprofessionals.org/usability_resources/surveys/2009salarysurvey_PUBLIC.pdf. Last accessed November 28, 2011.

59 Technology Transfer

Kevin M. Schofield

CONTENTS

59.1 INTRODUCTION

This chapter is intended to provide guidance on how to approach technology transfer of HCI-related research. Admittedly, there are many different perspectives one could take in looking at this often-difficult problem: researcher versus practitioner, industry versus government versus academia. There is also the extra, added dimension of whether the transfer is within the boundaries of a corporation or across corporate lines.

In this chapter, we will discuss tech transfer both internal and external to a company and then discuss commonalities across the two processes. Primarily, we will look at it from the perspective of the researcher, because in practice the burden is more on the researcher to justify the transfer and to make it work. Practitioners, however, will also gain value from reading this, as it will help them to understand the role of the researcher in technology transfer.

Although any technology transfer has its challenges, HCI tends to be a particularly difficult one. This is due to many factors, but the two main factors are that it is often more about abstract ideas than specific implementations and because we in the HCI community are still fighting the (wrong) impression that HCI is an afterthought and not the "real meat" of research and development efforts.

59.2 NEW FACE OF TECHNOLOGY TRANSFER

The last few decades have seen dramatic boom-and-bust cycles in the technology industry that, for a number of reasons, has changed the face of business, commerce, and information flow. Probably, the most earthshaking of those changes has been within the high-tech industry itself, as fitting for such a rabid consumer of its own technologies and ideologies.

The Internet, through its nearly ubiquitous connectivity to information and other people, has reshaped organizational patterns and forged a brand new set of relationships between researchers, designers, developers, manufacturers, marketers, and the other people involved in business and commerce. Those new organizations and relationships respect neither national nor corporate borders and bring us one step closer to the "friction-free" economy that was trumpeted in the mid-1990s.

Technology transfer is not what it used to be, largely because research and development is also not what it used to be. As Friedman (2005) pointed out, the world has flattened, and increasingly research and development can be done anywhere, with anyone, and for anyone. In lockstep, the relationships between corporate and academic R&D have also evolved; there are fewer and smaller standalone corporate research laboratories as many companies have chosen to "outsource" their basic research to academia. Even those companies that continue to invest in basic research are realizing that with fewer barriers to communication and collaboration, an "open innovation" model of joint R&D, as suggested by Chesbrough (2003), often makes sense to maximize the impact of internally generated ideas as well as to take full advantage of the most relevant externally generated ideas.

Government research, and funding, is also changing. Some of this is due to economic cycles; much is due to shifting political priorities. In Europe, for example, the Lisbon Agenda defines the European Union's desire to make Europe the most innovative economy in the world by 2009, and this drives their investment in research and development. In contrast, in the United States, despite a wide recognition that IT is driving economic advances, the major funding agencies (NSF, DARPA, and NIST) have reduced their support for academic research in computer science, with a built-in assumption that private industry should and will take up the slack. Somewhere in the middle, many countries (e.g., Canada, Australia) have programs where academic researchers who are funded by private companies can apply to the government for matching funding.

Still, as the National Research Council (2003) observed, successful technology transfer is usually a marathon, not a sprint, and may require years to reach completion. It also reflects a complex partnership between government, industry, and academia. What should we take away from this? That, clearly there is no one model for technology transfer from research into industry—there exists a whole spectrum. As mentioned previously, some governments fund research, while others do not, and some are in the middle. Likewise, some companies do their own basic research, while others outsource it, and some are in the middle. Some universities keep a death grip on their faculty's intellectual property; others, such as University of Wisconsin and University of Waterloo, have a policy that IP belongs to the inventor. Many are in the middle.

Consequently, there is no "playbook" that will explain exactly how to transfer a research result. Both researchers and practitioners will need to be very flexible and willing to adjust to the situation in which they find themselves. Tech transfer is, in the end, a business, and the best way to succeed at it is to think like a businessperson.

59.3 INTERNAL TECHNOLOGY TRANSFER

I am constantly surprised by the number of people who believe that technology transfer is some sort of Rube Goldberg machine,* where technology is inserted in one end of the contraption, strange things happen in the middle that usually involve people in uncomfortable and contorted positions, and then magically it pops out on the other end. Countless books and articles have been written (Lesko, Nicolai, and Steve 1998; Mock, Kenkeremath, and Janis 1993) in an attempt to document the perfect mechanical process for technology transfer. And yet, despite the fact that nearly everyone has had painful experiences trying to define a mechanical process for technology transfer, they still try to do it and complain bitterly when it fails (Butler 1990; Hiltzik 1999; Isaacs and Tang 1996; Singh 1993).

59.4 TECHNOLOGY TRANSFER AS A SOCIAL PROCESS

Tech transfer is not a mechanical or logistical process; it is fundamentally a social process. It succeeds when people build a relationship between the provider and the recipient that fosters trust and communication. Manning (1974) recognized that successful technology transfer centers on viewpoints and perspectives and fundamentally on communications. Foley (1996) spoke to this point most directly, that technology transfer is a "full-contact sport" that centers on the people.

Successful product organizations understand that risk is their mortal enemy. They work hard to proactively manage the risk in their development process or to remove the risk factors altogether. One of the most prevalent and difficult-to-manage risk factors is an external dependency, and let us be honest: an external dependency from a research organization looks about as risky as it gets. As long as your counterparts in product organizations think of a research organization that way, technology transfer is difficult at best and often outright impossible.

To succeed with technology transfer, we need to mitigate the risk or at least the perceived risk in the minds of the people we wish to receive our technology.

Up to this point, none of this is particularly controversial, but this is where the paths diverge. Many people will tell you that you succeed in mitigating the risk by creating well-defined, step-by-step processes through which you and your industry partner will enact the technology transfer. I argue that this approach fails more often than it succeeds, for two main reasons:

- The only experience that people in industry have with external dependencies is the occasional dependence on an external contractor or supplier to deliver a finished component ready for integration. They inevitably use this as the model for defining their tech-transfer process from research, and it is fundamentally incompatible. Research technologies are not finished components, and any product organization that expects a research group to deliver a finished component fundamentally misunderstands the role, expertise, and hiring practices of a research organization. Research groups almost never understand the development and test practices of a commercial product organization; even if they did, those practices vary so widely between organizations that past

* For those in the United Kingdom, that would be "Heath Robinson machine."

experience does not guarantee that they could successfully deliver a finished component. Moreover, it is not the goal of a research organization to develop technologies into finished components; its goal is to discover and prove solutions to previously unsolved problems. To do so, a research organization requires different skills and expertise and frankly different development and testing methodologies. Both research and product organizations need to understand this fundamental difference and embrace it as a way to complement each other's strengths, rather than ignore it and delude them about a theoretical capability that, practically speaking, is not there.

- People who do not understand each other cannot communicate and do not trust each other and cannot be expected to work cooperatively toward a shared goal, even within the best-defined process. The trust and communication must come first. If the two people on opposite sides of a table trust each other, then the two of them can accomplish anything; if they do not, they will never accomplish anything of value.

59.5 BUILDING TRUST IN STEPS

So then comes the catch: how does one build trust? By working side by side, of course! This means that one must start with the kind of activities that are initially low in risk, but high in communication, and build on one's successes to build more trust and overcome successively higher levels of risk.

Step 1 is to establish trust that one is an expert in the domain. Technical people, whether in research or in industry, are almost universally avid readers and understand the importance of staying up to date in their fields. But we all suffer from a lack of time to weed through the volumes of less-than-useful information to find the truly valuable nuggets of wisdom. If someone in a product organization expresses an interest in one's field, an offer to forward them a set of papers, articles, and books that represent the seminal reading is a great first response. Granted, doing a literature search is not glamorous work, but it fundamentally demonstrates a working knowledge of the domain and an ability to provide guidance. Equally important, it shows a healthy respect for the people's intellect and a flattering assumption that they will be able to read and digest the material. One of two things will happen. Either they will actually read the materials sent to them, in which case they have not only made an initial investment in seeing a technology transfer happen but have also been provided with great topics for further conversation, or they will not read the material and most likely conclude that it is simply easier to rely on the researcher as their expert rather than to become experts themselves. Either way, it is a foot in the door. They will ask endless questions as they try to decide for themselves what is within the realm of possibility and, more importantly, practicality. It is essential to ask them just as many questions to understand as completely as possible their constraints and to make clear recommendations on what they can expect to build.

Step 2 is to move from simply giving domain guidance on the state of the art to provide feedback on product-design plans. This involves offering to review specifications and providing timely feedback. Timeliness is critical—schedules are the rules of the game, and an ability to stay within their stated schedule reflects an understanding of the rules, an appreciation for their importance, and a commitment to the success of their project. This is also a critical test of a researcher's ability to think practically; in their distrust, they might expect suggestions of wildly impractical things that would have a negative impact on their schedule or require resources out of proportion the relative importance of the technology to the overall product. It is the researcher's job to show once again an understanding of their constraints and the value added to their team effort. Success will be apparent when a subtle but important shift happens: instead of the researcher asking to review their design documents, they will ask the researcher to review them.

Step 3 is a significant one indeed: when the clients ask the researcher to help write the specification for the product. Do not expect this to happen until there has been a clear success at Step 2 that has established credibility. But when it does happen, the product organization is making a loud and unambiguous statement that they now think of the researcher as part of the product team. This is an enormous step for a product group to take in their relationship, and it is a heavy responsibility to take on. At first, they will probably only delegate small parts, and often they might ask the researcher to co-write design documents. But regardless of the size of the assignment, the key to success is the same: whatever is designed must be easily buildable and testable. If there is any significant disagreement on whether the design can be built or tested, the product organization will not take the risk. Development organizations (at least the successful ones) are inherently conservative and will overstate the costs to build new technologies.* This is not only another test of whether the researcher understands their constraints but also equally whether he or she understands their development process. I encourage "aiming low" initially and looking for indications from the team that they would like to work together to design something more aggressive. If there is success in co-designing components, they will loosen the reins and delegate more responsibility (with more autonomy).

Step 4 is where one (finally) gets involved in implementation. It has taken enormous patience on the part of the researcher to get here, and there are still landmines everywhere. No two development organizations are alike; they all have different practices for creating, documenting, integrating, accepting, testing, and deploying new products. It is impossible to understand all of their processes, and it is

* Ironically, it has been my experience that development organizations tend to estimate incorrectly not in the new technologies, but rather in the incremental improvements to legacy components, and particularly in the "integration" work in making multiple components work together.

very unlikely that they will all be written down; yet, every one of them has an opportunity to break form and cause a rift. I would encourage asking the group manager how they bring a new employee into the group and what training and mentoring that person would go through; further, see if there are opportunities to take advantage of such a process to help to get up to speed. If one has made it this far, the product team wants to see a success as much as the researcher does. Because one's success is the other's success, the product team will be very reasonable about doing things to help themselves be understood by the researcher, especially if it is clear to them that that is the goal. All development groups fall in love with their own processes, and one can earn their cooperation by showing equal respect for those processes, no matter how silly they might seem to an outsider.

The key to success in a development process is to realistically promise and over deliver. Set rational expectations; they should think that you can carry your own weight, but not that you are God's gift to engineering. Be honest about the readiness of your technology in as crisp terms as possible; Speser (2006) suggests terminology for Technology Readiness Levels that represent how far away from market introduction the technology is currently. Promise metrics for work and for the technologies created that are achievable but not overly aggressive, and then exceed those metrics. By doing that, it is possible to fully gain their trust and move on to discuss with them more aggressive technology transfers.

It is important to note that in this "pyramid" of sorts that we are building with increasing levels of risk and corresponding trust, it is possible to peak at any level. For instance, if the researcher does not have the development skills to co-develop components with a real product team, then do not try to do it! By all measures, every step in this process can be considered technology transfer. Product organizations need knowledge, understanding, and ideas about technology just as much as they need finished technology components; they need to understand what cannot be built just as much as what can. And, most importantly, they need researchers to tell them honestly what they are and are not capable of doing for them. Even without ever delivering a finished component to a product organization, one can still have a litany of technology-transfer successes for which the product organization will sing praises. It is more important to proceed in measured steps built upon past successes and build the trust and the lines of communication that will guarantee future successes.

59.6 THINKING FOR THE FUTURE

It is also important to be thinking to the future—to be thinking about what comes next. There is always a desire to simply throw a technology over the wall and then to move on to the next research project, but this is unrealistic. It never really works that way, and even if it did, "throwing it over the wall" would end the relationship and any opportunities for future technology transfer; the ongoing relationship after the transfer is an opportunity to carry on a dialogue about the next great technology.

From the product–organization perspective, there is rarely a "clean" way to integrate a component. The overwhelming majority of development work is revisions to existing products; very rarely are new products started. Revision work means that new components need to be integrated into an existing legacy framework; this usually requires development work on both sides of the integration to ensure the optimal match.

59.7 CONSIDERING HCI IN THE TECH TRANSFER PROCESS

As if this was not difficult enough, applying this approach to HCI-related technology transfer introduces its own challenges. One can read The Psychology of Human–Computer Interaction (Card, Moran, and Newell 1983) and learn that at a very fundamental level, a set of scientific principles holds very broadly. However, we in the HCI community have also learned that interactive systems must be designed within the context of a particular task and human and that this very fact makes it tricky at best, and misleading or impossible at worst, to try to generalize specific designs to other contexts. Even with the best of intentions and the most thorough usability testing, there are no clear guidelines about how much of an HCI-related research technology can actually be transferred and particularly for integration into an existing product. So "throwing over the wall" is especially difficult, as it calls for potential redesign, as well as further development and integration, test, localization, support, and operations (it is increasingly a service world, after all).

In the traditional view of technology transfer, lack of a clean handoff would be fatal. In the "relationship" view, however, this is an opportunity to build a working model that lets you overcome the challenges and work side-by-side with a product organization to guide the transfer of your work.

Beyond simply moving up the pyramid, one can do other important things to deepen the relationship. Scheduling regular "maintenance" conversations can help to maintain communication channels and to keep abreast of activity in the product organization; it is also an opportunity to continue to update them on progress on new work. The relationship can also be used to improve the researcher's own work by learning about critical real-world issues. Good product organizations have a wealth of information about their customers; by using their access to real customers (and aggregate information about them), the relevance of research activities can be improved. It is an opportunity to ask key questions, learn about critical product strategic direction, conduct user studies, and find out what difficult HCI and technical issues are about to become critical issues for real customers. This is the golden opportunity to get out of the ivory tower.

As an aside, it is worth pointing out that HCI has its own value-add in the tech transfer process helping to quantify improvements. Often it is important, when there are competing technologies, to demonstrate the superiority of your technology. HCI's processes to measure quantitative differences in ease of use and "time on task" can be very helpful in these cases.

59.8 EXTERNAL TECHNOLOGY TRANSFER

Technology transfers outside the boundaries of a legal entity, regardless of whether they originate in a government agency, academia, or an industrial lab, are almost by definition cleaner types of transfers. This means that one will be participating in a transaction involving the sale or licensing of technology, or contracting to provide some service, or both. So, the first order of business is negotiating the "deal."

59.9 KNOW WHAT YOU ARE SELLING

It is critical to know and understand what is being sold: outright ownership of intellectual property or a license to it? Is it a complete solution that has been developed or just pieces? Those pieces might include any or all of the following:

- User interface design
- System specifications
- A working prototype
- An actual implementation, tested to some level of quality
- The source code for a software implementation
- Copyrights
- Patents
- Working time as a commitment to support ongoing productization
- A running service that you host

59.10 KNOW WHAT YOU ARE NOT SELLING

It is equally important to know what is not being sold. If one would like to continue this work, he or she will need to make sure to preserve rights and ownership to continue that work. Otherwise, one could very well put oneself out of business by selling complete ownership to a valuable asset or by signing an exclusive license, which precludes licenses with any other company.

59.11 KNOW THE PEOPLE INVOLVED

Deals inevitably involve lawyers as well as what are known as "business development" people: those whose job are to negotiate deals that further the business interests of their employers. One can assume that the company negotiating will have both business development people and lawyers; it makes good sense for the researcher to have them too. This is a situation where we need to put our pride and high-minded notions about doing business "on a handshake" aside. While the overwhelming majority of companies are not in the business of stealing from people like us, and will not try to do so, to get the most value in return for what is being offered one needs someone on his or her side who understands what is customary in intellectual property deals and how to negotiate for it. Even a brilliant and excellent debater who does not know what to reasonably ask for is at a serious disadvantage. The bottom line: find someone with good business development skills and experience to negotiate the deal.

Likewise, once the terms of a deal have been negotiated, one needs a lawyer to write it up and make it legally binding. Do not even think about self-representation in the drafting of an intellectual property agreement; the laws are changing too quickly (which is not the fault of the lawyers) for a researcher to understand which ones apply to the situation and should be factored into the drafting of an agreement. Mock, Kenkeremath, and Janis (1993) set out the basics of existing laws and how they relate to technology transfer, though the details have changed substantially since then. Many good books exist to help one to get educated on current intellectual property laws, although that is a poor substitute for a competent attorney skilled in the current practice. The Association of University Technology Managers (http://www.autm.net) also provides a wealth of resources to its members on a number of issues related to technology transfer and intellectual property.

The key to success is to understand the defined role of each of the three people on a negotiating team: oneself, the business-development person, and the lawyer. The researcher's role is to be the technical expert and to place a value on the work as well as what the people on the other side of the table are offering in return; one's role as "client" is to decide what is needed to be successful and what additionally is desired but negotiable. The role of the business development person is to take the articulated needs and desires and try to structure the terms of an agreement that will work for both parties. He or she understands business risks and will help the team members to understand them and make informed decisions about how much risk can be tolerated. The role of the lawyer is to take the terms and write them down in words that both parties understand and that can be interpreted under the law to protect the client's interests. The lawyer also understands and can articulate the legal risks; laws are often subtle and ambiguous things that can be interpreted in many ways (in fact, nations have an entire branch of government that does nothing but interpret the laws). Any contractual obligation runs the risk of being interpreted in a way other than how it was intended, and a lawyer can help the team members to understand how likely that is, based on the language of the law and similar previous cases where the law has been interpreted by the courts. Just as there is always business risk, there is always legal risk, and in the end, it will fall upon the researcher to make the decision as to whether the risk is acceptable.

It is also important to understand who plays these roles on the "buying" team. Speser argues that the most important role to understand (and in particular who is playing that role) is that of the decision maker, who at the end of the day will actually decide whether to go ahead and acquire a technology. It most likely will not be someone in the legal department or someone in their internal research department, who might even see you as unwanted competition. It most likely will be the person in the development team who actually plans to use the technology, as that person is the one who will need to find the money to purchase it as a part of running their business.

Where do deals go wrong? In my personal experience, they often go wrong when these three roles become confused and when the business development person and the lawyer start to make the key decisions. The researcher must live with the result of the deal, not them. On one end of the spectrum, there is no such thing as a risk-free deal; on the other end, even a high-risk deal could be worth doing if the reward is also high enough. Those decisions are the client's, not theirs, and the researcher as client should insist on making them.

There are many good sources for both business development people and lawyers. Venture capitalists will often have a "short list" of ones they trust to do their business transactions. The Chamber of Commerce for a local area can also provide recommendations and often will track complaints registered against specific ones. Many regions also have associations of entrepreneurs, inventors, and small-business owners, with great resources to draw from.

Of course, if one already works in a research lab, the institution most likely has a technology-licensing office that will negotiate and draft deals on the researcher's behalf (and are likely required to do so if the researcher's employment agreement assigns ownership of inventions to the employer). In that case, one will still need to stay involved to make sure that the researcher's needs are met in whatever deal is negotiated.

59.12 CRAFTING A DEAL

The most difficult process of negotiating a deal is crafting an arrangement that meets the needs of both sides. I have seen that many negotiations take much longer to conclude, and in many cases fail to conclude successfully, because either or both sides did not bother to try to understand the other side's business needs. Business partnerships are always about finding a way to help both parties be more successful. Speser goes as far as to suggest that deal making is the search for a Nash equilibrium whereby all parties have more incentive to stay with the deal than to change their tactics. The best way to do that is to understand what one's prospective partner's business is about and likewise to share enough information about one's own business, openly and honestly, so that together a combination can be found that works for both sides. Find out everything available about a partner's current business situation:

- Their revenues and profits
- Their competition
- Their most important customers
- What customers are saying about their product
- Where they say they want to take their business in the future
- The problems and challenges they are facing

This essential information will guide you to deeper insights on how to offer terms that will be seen as valuable to a potential buyer or licensee. Steinberg (1998) described his experiences in negotiating deals and his own well-known and well-respected philosophy for how to structure deals makes good business sense for all parties.

Companies will pay for the value that can be delivered to them. They will pay a certain amount of money to make an even larger reduction in their costs (because in the end they save money). They will of course pay to help themselves make even more money. Finally, they will pay if one solves a problem for them. As a deal is structured, try to cast it in terms of what it does for them; those are terms that they can understand and, more importantly, quantify in a valuation.

Take it further. Help the buyers in any way possible to place a value on what is brought to the table. For example, conduct a user study on their existing product and another showing how the technology being offered will improve their product (if it happens to address a key customer complaint, all the better, and that should definitely be brought to their attention). During the negotiations, show them a smart, effective professional who can work with them. This does not mean that you need to negotiate hard to the very last item; contrary to popular belief, the tough negotiators are not always the most respected, and in fact, they are often the ones that create their own reputation for being difficult to work with. It is much more important to demonstrate an understanding of the buyers and an ability to speak their language, as well as willingness to help them make the case to the decision makers in their organization's senior management.

59.13 CONFIDENTIALITY AND NDAs

One large challenge in negotiating deals is the issue of confidentiality, which usually rears its head first in the often-dreaded nondisclosure agreement. Nondisclosure agreements are signed before revealing confidential information to ensure that the information would not be disclosed to competitors. That part is a good business and a natural, noncontroversial part of good-faith bargaining—we all need to be able to keep secrets. The difficult part of nondisclosures is the issue of residuals: by looking at confidential information, I learn things and then I carry that learning around in my head for the rest of my career. What am I allowed to do with that information in my head, and who in fact owns it? From the discloser's point of view, one wants to make sure that someone cannot use his or her own confidential information to compete. From the other side, it is impossible to know exactly what will be disclosed, or what business opportunities are going to come one's way tomorrow, so it is deeply problematic to sign away the ability to enter certain businesses simply for the privilege of looking at confidential information. Both sides sound very reasonable, and they are, which is why NDAs are no trivial matter and often become the stopping point in negotiations.

Whenever possible, try to complete as much of the negotiations as possible without entering into an NDA, because it simplifies matters and prevents the trust issues from rising to the surface too early. The downside is that this makes the early negotiations a precarious dance, where a researcher needs to show the other party enough to convince them that the technology is real and solves their problem, without giving away key secrets. There are things that still can be done: tell them "what" it does, instead of "how" it does it, and show them the system working. The goal is to make them crave it enough that they will want to sign an NDA under the researcher's terms to complete the technical due-diligence required for them to close the deal.

It is critical to think this all through before getting to the negotiation table—these are never decisions to make under time and social pressure. It is also critical to realize that the whole issue of confidentiality and NDAs is one more business risk; admit it and decide for yourself whether (and when) the potential reward outweighs the risk. This is one of the clear cases in which a lawyer will be extremely conservative and protective and describe in great detail everything that could be lost by entering into an NDA (or by showing technology without an NDA). But, in the end, the decision is the researcher's.

59.14 GOING TO THE NEGOTIATING TABLE

Negotiating a deal is probably the most hyped and feared part of this whole process; perhaps, we have all had too many nightmares about slick car salesmen tricking us into paying too much for too little in return. The reason we fear car salesmen is that the salesman has all the information about what the car is really worth and shares none of it with us; we are forced to blindly trust him, and many of us do not.

Steinberg (1998) once again shared his wisdom on a sound and ethical approach to negotiation in his "twelve essential rules for negotiating." Not nearly as ambitious as Steinberg, I have only three basic rules for negotiating:

1. I obtain as much information about each side's position as possible before arriving at the table.
2. I have a list of what I really need to succeed and a separate list of what I want to have in addition. I hold firm on my needs, and I am willing to compromise on my wants.
3. I always negotiate a deal in which both sides win.

Understanding a potential business partner's position is critical to negotiating success for a number of reasons. First, it tells what they are looking for. Ask the same list of questions we discussed with internal technology transfer: Who are their customers? What are those customers saying about their products? Who are their competitors? What is the company looking for that will help them to be more successful? What are their strengths and weaknesses? Second, and very much related, it tells how the buyer will value what they are offered. Successful business deals involve an exchange of value that both sides view as fair and equitable, but value is of course relative to the company and its context. Understand that in order to find an equal trade. Reading the company's annual report is an excellent source of information about a company (if it is a publicly traded company). Reading news articles and competitive reviews also provides invaluable information about the business pressures the company is under, as well as the assets that they bring to the table.

Having the list of the things that one really needs and the things that one wants in addition is a valuable step in preparing for negotiation. I have seen many people come to the table unwilling to compromise on anything; they ask for too much in the beginning and believe strongly that compromising on anything is a sign of weakness. Negotiations like those always take longer than they should and are very frustrating. In some cases, they spend more money in lawyers' fees for fruitless negotiations than the value of the small items on which they refuse to compromise. I recommend starting with a basic negotiation on the core needs of both sides; not only does that keep you focused on the heart of the deal but it also tends to simplify things just by taking all the peripheral items off the table. Once the heart of the deal is done, and both sides feel comfortable that they can be successful because they are getting what they need, adding additional pieces is much easier with a lower stress level and a structure in place.

Remember to be honest with yourself as you detail all this information, because lying to yourself is the surest path to failure. Assume that they will have an accurate valuation of what is brought to the table (regardless of whether they are willing to tell you what it is), so insisting that something is worth more than its real value is foolish. Understand one's own strengths and weaknesses, be honest about what is needed to be successful, and do not promise things that cannot be delivered.

All this brings us to the last rule: always negotiate a win–win. Always negotiate a deal in which both sides feel that they are receiving what they needed and can be successful. Beyond the obvious ethical reasons for doing this, there is also the very practical consideration that the two parties will need to continue to work together. Many people go to the negotiating table believing that signing a deal is the end of the process, when in fact it is just the beginning. Once the paperwork is signed, a relationship that essentially lasts forever begins between the parties. This relationship often makes itself known in unanticipated ways; for example, if one is licensing a patent to a company, the company has a vested interest in ensuring that the patent maintenance fees continue to be paid to the PTO to keep the patent valid and will want to have regular information to confirm that the payments are being made. Almost every deal one can imagine, no matter how cut-and-dry, has some aspect that will require communication between the parties on an ongoing basis after the deal is done. If one licenses a technology to a company, the company may be sending royalty checks, and the licensor may want some way to audit their sales to ensure that they are accurately paying. They, in turn, may want technical support, including important bug fixes and updates. They may additionally have negotiated the option to license future upgrades, and as a paying customer, they will likely want to provide their input on features and enhancements to the technology that would be of most help to them.

59.15 EMBRACING THE RELATIONSHIP

By admitting from the beginning that there is an ongoing relationship, one can embrace this notion and turn it to an advantage. In fact, I encourage its use to build a future revenue stream—and in light of the increasing attention to Chesbrough's "open innovation" model, this is a likely outcome indeed. A researcher can build design and consulting services into the deal, which is particularly helpful for HCI-related technology transfer since, as we discussed earlier, they often need reworking to fit into a larger context. One can use the ongoing relationship as a "foot in the door" to be able to offer future sales and deals as new technologies are developed. In fact, viewed as a "strategic partner," one might even want to offer them the right of first refusal on future offerings, as a way of demonstrating a commitment to them. Finally, as in contemplating future growth of the researcher's business, one may need additional sources of funding, and a partner who has a vested interest in the researcher's success can be a great source for funding. Even if none of this is true, assume that one day there will be a need for a good reference or recommendation from them; that alone is reason enough to want to have a great ongoing relationship.

59.16 COMMONALITIES FOR INTERNAL AND EXTERNAL TECHNOLOGY TRANSFER

59.16.1 Intellectual Property

One of the issues common to both internal and external technology transfer is intellectual property (IP). I am not a lawyer, and so I obviously cannot give legal advice on how to protect intellectual property or how to treat others' IP. What I can do is point out some places where the IP issues get thorny and make some business recommendations about how to deal with them.

Anything received from a third party may carry restrictions on how it may be used and, more importantly, whether it can be redistributed in its original or modified form or combined with some other components. This includes libraries of software routines, data, copyrighted works and designs, and patents. These restrictions can come from explicit license agreements that accompany the third-party components, or they could come from any of a number of different laws, including patent, copyright, trademark, trade secret, and export. Any time a third-party component is used in one's work, a business risk of constraining the ability to transfer the work to another party arises, because either one does not possess the right to do so or the rights that are possessed are not sufficient to the needs of the party that wants to license it. This question fits into the larger scheme of what it traditionally called the "build or buy" decision: whether it makes more business sense to build something oneself or to buy or license it from a third party.*

I strongly recommend that, whenever possible, the IP issues should be dealt with at the time a third-party component is acquired, rather than waiting until an opportunity to transfer it. This accomplishes two things. First, it allows one to negotiate and make business decisions about acquiring the component before there is a commitment and dependence on the component built into your technology; once the dependency is there, the "switching cost" is much higher for moving to an alternate and one could be forced to pay a much larger licensing fee than before. Second, it simplifies the tech-transfer process. Any company worth its salt will perform a "due diligence" on the technology before it closes a licensing deal. Part of that will be an analysis of who really owns the technology or whether the licensor has acquired the right to further license it. In essence, one will need to prove the right to license one's work to the company. And do not be surprised if the company also asks the researcher to "warrant" the work—to guarantee that he or she has the right to license all of it to the company and that right will be defended in court if necessary. The bottom line: clear it up front, and save a lot of trouble later.

59.17 IT ALL COMES DOWN TO THE RELATIONSHIP

The most significant common aspect of internal and external transfer comes back to the notion that we began with: technology transfer is a social process that succeeds or fails based on the relationships that have been built. Tech-transfer partners need to trust each other and that trust is built with communication and follow-through. Researchers should understand clearly and honestly what value they bring to the table at the various stages of the relationship, and they should make it their business to know how their partners see and value what they bring. After all, business is fundamentally the exchange of value between partners who need each other's competencies, and the truly successful companies are the ones that build relationships that last across a continuous series of business transactions.

When beginning a technology-transfer effort, assume that what is started that day is the beginning of a working relationship that will last forever. Build the relationship from the ground up with the expectation of ever increasing levels of cooperation and trust that will allow the partnership to take on ever more challenging technology transfers, in whatever manner is most appropriate to the needs of the business.

There are far too many stories of companies who have struggled with the transfer of technologies that could change

* It is important to point out, though, that building from scratch does not necessarily mean an automatic escape from third parties; for example, one can still violate someone else's patent even with code written from scratch.

the world—and failed. As Buderi (2000) and Freidman (2005) described, the next chapter of this story is being written now. By rewriting the rules to focus on the social side of the process, we can ensure that our best work will see the light of day and this story will have a happy ending. The good news is that there are more companies open to technology transfer today than at any time in the history of the industry, and the opportunities are there—on both sides—for us to take.

REFERENCES

Buderi, R. 2000. *Engines of Tomorrow: How the World's Best Companies are Using their Research Labs to Win the Future.* New York: Simon & Schuster.

Butler, K. 1990. Collaboration for technology transfer—or "How do so many promising ideas get lost?" In *Proceedings of the CHI '90 Conference on Human Factors in Computing* Systems, 349–51. New York: ACM.

Card, S. K., T. P. Moran, and A. Newell. 1983. *The Psychology of Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Chesbrough, H. 2003. *Open Innovation.* Boston, MA: Harvard Business School Press.

Foley, J. 1996. Technology transfer from university to industry. *Commun ACM* 39(9):30–1.

Friedman, T. 2005. *The World is Flat.* New York: Farrar, Straus, and Giroux.

Hiltzik, M. 1999. *Dealers of Lightning.* New York: Harper Collins Publishers.

Isaacs, E. A., and J. C. Tang. 1996. Technology transfer: So much research, so few good products. In *Proceedings of the CHI '96 Conference on Human Factors in Computing Systems*, (2), 155–6. New York: ACM.

Lesko, J., P. Nicolai, and M. Steve. 1998. *Technology Exchange in the Information Age.* Columbus, OH: Battelle Press.

Manning, G. K., ed. 1974. *Technology Transfer: Successes and Failures.* San Francisco, CA: San Francisco Press.

Mock, J. E., D. C. Kenkeremath, and F. T. Janis. 1993. *Moving R&D to the Marketplace: A Guidebook for Technology Transfer Managers.* Falls Church, VA: Technology Prospects, Inc.

National Research Council. 2003. *Innovation in Information Technology.* Washington, DC: National Academies Press.

Singh, G. 1993. From research prototypes to usable, useful systems: Lessons learned in the trenches. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 139–43. New York: ACM.

Speser, P. 2006. *The Art & Science of Technology Transfer.* Hoboken, NJ: John Wiley & Sons.

Steinberg, L. 1998. *Winning with Integrity*. New York: Villard Books.

Part VII

Emerging Phenomena in HCI

60 Augmenting Cognition in HCI

Twenty-First Century Adaptive System Science and Technology

Kelly S. Hale, Kay M. Stanney, and Dylan D. Schmorrow

CONTENTS

60.1 INTRODUCTION

In 1960, J. C. R. Licklider had a vision for a "Man–Computer Symbiosis" in which the human and computer, although dissimilar from one another, would live together in an intimate association, producing increased handling and new ways of processing information (Licklider 1960). Over the past few decades, several attempts to realize this vision have been made by interactive system developers, but each time it has eluded them. This was likely due to the insufficiency of technology and computational power, but also to the need to mature several fields of basic science necessary to understand how human–machine symbiosis might be produced. A more thorough understanding of human brain functioning and what guides behavior during human–computer interaction (HCI) has been a continuing missing requirement in the ability to enable true human–machine symbiosis.

Fortunately, in the 1990s period known as the "Decade of the Brain," the National Institutes of Health (NIH) and other federal funding agencies invested heavily into advancing neuroscience in order to understand the basic scientific aspects underlying the brain to include human cognition and behavior. What emerged are advanced neurophysiological assessment tools and a wealth of knowledge regarding specific brain areas that could be attributed to particular cognitive and behavioral functions. As brain activity measurement tools, such as electroencephalography (EEG), magnetoencephalography (MEG), positron emission tomography (PET), functional magnetic resonance imaging (fMRI), and functional near-infrared imaging (fNIR), continued to advance into the year 2000, the idea that one could possibly begin to capture brain functioning in real time as users performed real-world tasks came to fruition.

Seeing the promise of such capabilities, in 2001, the Defense Advanced Research Projects Agency (DARPA) began funding a program based on what is now known as the scientific field of augmented cognition (AugCog) (Kollmorgen 2007; Schmorrow and Kruse 2002, 2004, 2005; Schmorrow and Reeves 2007; Schmorrow et al. 2005). Although the field of AugCog has had many predecessors (e.g., DoD-funded programs in Biocybernetics, Learning Strategies, and the Pilot's Associate Program; NASA-funded adaptive automation efforts [Freeman et al. 1999; Pope et al. 1995; Prinzel et al. 2000; Wilson, Lambert, and Russell 2000]), it was not until DARPA's AugCog program, where multidisciplinary teams of neuroscientists, cognitive scientists, computer scientists, software engineers, and human-centered system designers were brought together, that the potential of real-time, closed-loop symbiosis between human and computer was fully realized. Under this effort, human state was captured and analyzed in real time and used to adapt computer interface and procedures with the goal of optimizing human performance. Since the success of the AugCog program, there have been numerous follow-on programs (e.g., DARPA's Improving Warfighter Information Intake Under Stress [IWIIUS] program; DARPA's Neurotechnology for Intelligence Analysts [NIA] program, and the Intelligence Advanced Research Projects Activity (IARPA)'s Tools for Recognizing Useful Signals of Trustworthiness [TRUST] Program) and cutting edge research completed to advance AugCog systems for training (Craven et al. 2009; Pojman et al. 2009; Schnell et al. 2009), and operations (Carroll et al. 2010; Hale et al. 2007; Hale et al. 2008; Kruse and Schulman 2006). Today, the field of AugCog is aimed at substantially improving human–system interaction by using proactive systems that (1) detect and gauge a user's cognitive state (operator functional state [OFS]) in real time using a combination of diagnostic behavioral, physiological, and neurophysiological sensors, and appropriate data classification methods, (2) diagnose periods of nonoptimal performance (e.g., operator overload and repeated evidence of training error), and (3) mitigate the HCI experience via dynamically adaptive strategies with the goal of optimizing human performance (Schmorrow et al. 2005).

This chapter summarizes the latest science and technology (S&T) advancements from the field of AugCog, identifies applications of AugCog technology, lists lessons learned from the first decade of AugCog advances, and outlines future directions for AugCog research and development.

60.2 NEUROPHYSIOLOGICAL AND BEHAVIORAL TECHNOLOGIES FOR ASSESSING USER STATE

HCI researchers and practitioners must be able to rely on tools and techniques that allow them easy access for noninvasively observing and assessing users while interacting with human–computer systems. A critical component in gaining cognitive-state data is the use of behavioral, physiological, and neural sensors that can be appropriately combined to measure cognitive load in order to sufficiently characterize the cognitive state of users while they interact with a computer-based system in real-world settings. Here, cognitive state is defined as OFS—the moment-to-moment dynamic and functional capabilities (e.g., capacity and bottlenecks) of the human brain and/or a condition that has a causative/moderating/predictive relationship to a performance variable (Reeves, Schmorrow, and Stanney 2007a). This section outlines technologies that have been implemented in AugCog systems over the past 10 years that meet the requirements of (1) sensitivity to different brain states and/or processes, (2) reliability, and (3) practicality in fielded use (Gratton, Kramer, and Fabiani 2008). For a more thorough review of sensor technologies that encompasses technologies that are today cumbersome and nonportable, but which show promise in the future for capturing cognitive state in real time, readers are directed to the NATO report on OFS Assessment (Wilson and Schlegel 2004).

60.2.1 ELECTROENCEPHALOGRAPHY

EEG records electrical activity produced by the brain via sensors placed on the scalp (Figure 60.1) with high-temporal resolution (milliseconds). The activity recorded is a summation of millions of individual neuronal synapses, which limits the spatial accuracy of EEG in determining distinct, localized activity compared to other measurement techniques (Gratton, Kramer, and Fabiani 2008). However, despite its limits in spatial accuracy, EEG in closed-loop systems has been used to capture subtle shifts in cognitive function and cognitive processes such as sensory memory, working memory, attention, and executive function (as reported by Morrison, Kobus, and Brown 2006), alertness/vigilance (Duta et al. 2004; Jung et al. 1997), engagement (Freeman et al. 1999), cognitive load (DuRousseau 2004), and workload (Pleydell-Pearce, Whitecross, and Dickson 2003; Smith et al. 2001). Some specific AugCog gauges that have been developed using EEG include the eXecutive Load Index (XLI) gauge (DuRousseau 2004), which utilizes EEG signals to allow measurement of patterns in tightly coupled cortical networks tied to an individual's allocation of attentional resources as a user's cognitive state changes in response to conditional task load (Kobus et al. 2005), the Engagement Index, which is a ratio of EEG power bands (beta/[alpha 1 theta]) (Kobus et al. 2005; Freeman et al. 1999), and workload, engagement, distraction, and drowsiness gauges, where probability scores on each of these scales are provided on a

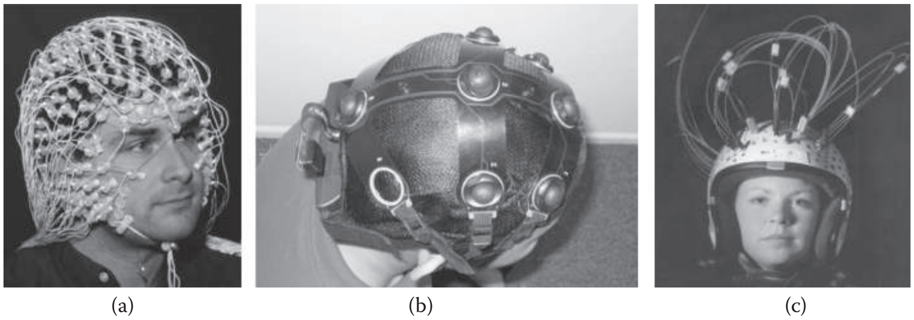


FIGURE 60.1 Electroencephalography (EEG) headsets used in AugCog systems developed by (a) Electrical Geodesics, Inc. (b) Advanced Brain Monitoring, Inc. (c) Beckman Institute.

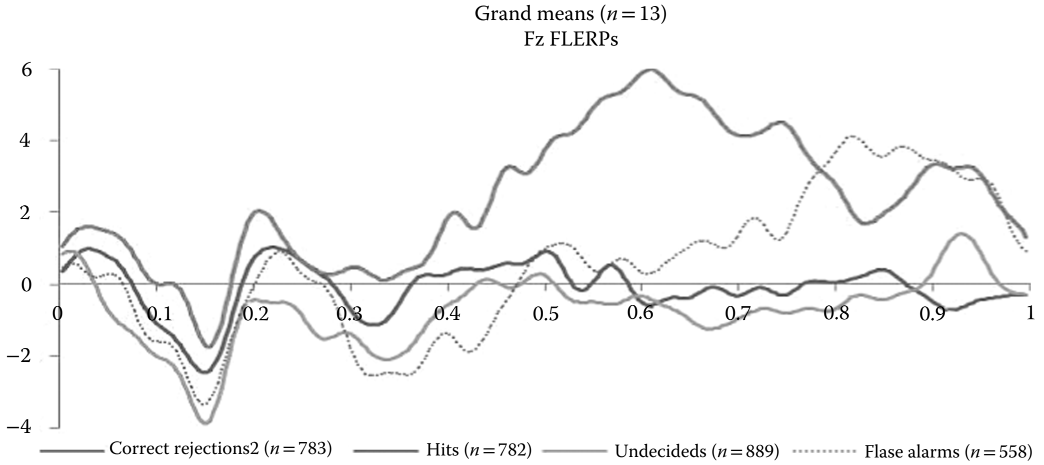


FIGURE 60.2 Example of fixation-locked, event-related potentials (FLERPs) from the Fz scalp site for distinct fixation classifications (correct rejections, hits, undecideds [misses that were later classified as hits] and false alarms).

second-by-second basis (Berka et al. 2007). The current state of EEG suggests it is effective in measuring the general level of arousal of the brain and has a good signal-to-noise ratio, low cost, low invasiveness, and relatively good portability (Gratton, Kramer, and Fabiani 2008).

Although the above gauges relied on changes in EEG band activity, a second approach to analyzing EEG is via event-related potentials (ERPs), which are thought to reflect "the cognitive processes that underlie task processing and responses" (Stanney et al. 2009). Using this technique, EEG signals are time stamped to an event (e.g., system event, behavioral response, and physiological event), and distinct patterns of response are used to distinguish various states, such as categorizing events as interesting versus uninteresting (Hale et al. 2008). Examples of EEG/ERPs include the P300, elicited by attended stimuli, and the error-related negativity (ERN), elicited when one is aware of making an error (Gratton, Kramer, and Fabiani 2008). Luu, Tucker, and Stripling (2007) found distinct EEG ERP components (specifically the medial frontal negativity [MFN]) that changed over time as participants learned a task. Research also indicates the feasibility of using EEG/ERP to differentiate between correct responses (i.e., hits and correct rejections) and highly biased responses (e.g., false alarms and misses) (Vogel and Luck 2000; Yamaguchi, Yamagata, and Kobayashi 2000; Sun et al. 1994), thus supporting the potential for using neurotechnology for enhancing operator performance.

Although most ERP work has utilized system events as triggers, more recent work has used eye fixation data to identify an analysis window to identify fixation-locked ERPs (FLERPs) using complex imagery (Hale et al. 2008) (Figure 60.2). Visual inspection of EEG ERP (time locked to image onset) and FLERP patterns showed distinct template signatures associated with a number of fixation classifications (e.g., hits, misses, correct rejections, and false alarms). Preliminary evidence suggests that the waveshape characteristics of FLERPs resemble those of stimulus-evoked ERPs, and a 1-sec analysis window from fixation onset indicated distinct signatures for correct rejections in addition to hits and false alarms (based on signal detection classification; Hale et al. 2008). This work suggests that FLERPS could be used during image analysis to increase efficiency and effectiveness.

The challenge in EEG/ERP classifiers is developing algorithms that can accurately and reliably determine classifications on a single-trial basis in near real time. While group classifiers would result in generalizable systems across individuals, all existing single-trial classifiers today utilize an individualized model to achieve accurate results (Parra et al. 2005; Mathan et al. 2006; Sajda, Gerson, and Parra 2003).

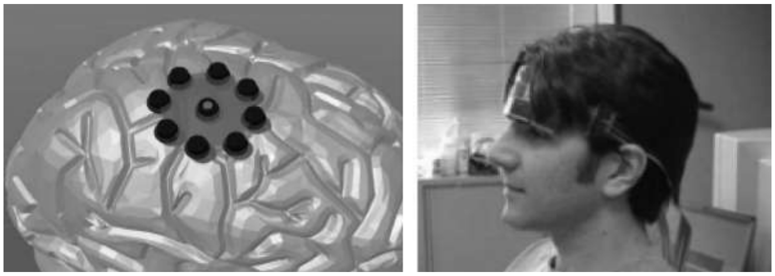


FIGURE 60.3 Functional near-infrared imaging (fNIR) sensor developed by Archinoetics, Inc.

60.2.2 Functional Near-Infrared Imaging

fNIR technology measures blood oxygenation and volume changes in the brain relative to where the optical sensors are placed on the head (Kobus et al. 2005) (Figure 60.3). fNIR has been shown to be an effective tool for diagnosing cognitive activity associated with spatial and verbal working memory, given its often known right- and left-hemispheric separability, respectively (Smith and Jonides 1998). fNIR sensors have been used to measure a variety of cognitive states in closed-loop systems, such as sensory memory and working memory (as reported by Morrison, Kobus, and Brown 2006), general workload (Izzetoglu et al. 2003), and loss of concentration (Izzetoglu et al. 2005). fNIR provides almost continuous estimates of changes in blood flow with a spatial resolution of a few centimeters, providing a relatively low cost and low invasiveness solution (Gratton, Kramer, and Fabiani 2008), however, there is time delay on the order of seconds associated with this sensor, due to the relatively slow nature of oxygenation changes within the brain (several orders of magnitude slower than EEG; Gratton, Kramer, and Fabiani 2008). In addition, fNIR cannot accurately measure deep brain activity, which may limit its diagnostic capacity (Gratton, Kramer, and Fabiani 2008). Even with such limitations, the relatively low cost and compatibility with other systems make fNIR a valuable addition to AugCog sensor suites (Stanney et al. 2009).

60.2.3 Electrodermal Activity/Galvanic Skin Response-Based Arousal and Cognitive Workload Gauge

The sympathetic nervous system modulates fluctuations in the electrical resistance of the skin, known as electrodermal responses (EDRs; commonly referred to as galvanic skin response or GSR) (Stanney et al. 2009). Although GSR is typically measured using the palm of the hand or the fingers, the soles of the feet and toes have an equal abundance of eccrine sweat glands. Thus, the feet and toes offer an optimal site for sensor placement because they will not intrude on a user's ability to interact with input devices that require hand operation (e.g., keyboard and mouse). GSR levels have been linked with variations in attention and working memory (as reported in Morrison, Kobus, and Brown 2006); cognition, attention, and emotion (Critchley 2002); engagement (Mandryk 2005); emotional response (Bradley, Moulder, and Lang 2005); and anxiety and stress (Healey 2000). This metric has been included as one of a suite of metrics to assess cognitive state in AugCog systems (Kobus et al. 2005).

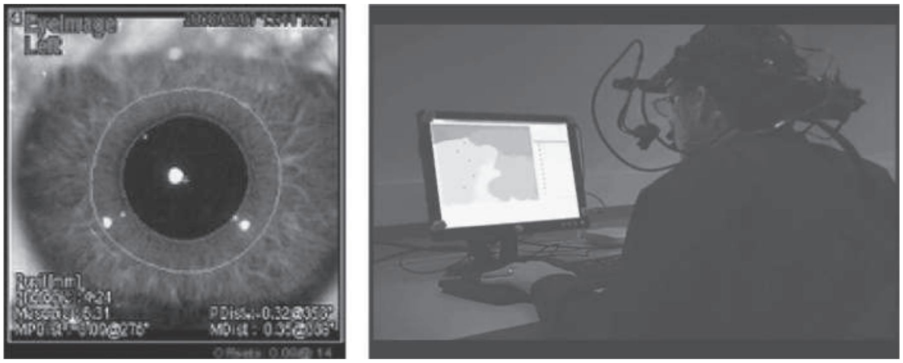
60.2.4 Heart Rate Variability

Heart rate variability (HRV), which captures variation in cardiac interbeat intervals, has been used to estimate arousal and workload (Kobus et al. 2005). Jang et al. (2002) used HRV to determine levels of arousal and engagement, as have Hoover and Muth (2004), who developed the arousal meter that derives autonomic arousal from the cardiac interbeat interval derived from an electrocardiogram (ECG) at 1-ms accuracy. This particular gauge provides three levels (low, medium, and high), which increase or decrease with respective increases or decreases in autonomic arousal. The arousal gauge has been used to track decrements in performance due to low-arousal states and is thus appropriate for assessing human information processing (HIP) bottlenecks related to attention (e.g., divided attention and vigilance effects).

A combination of heart rate and variability can discriminate emotional states (Jang et al. 2002; Lee et al. 2005) and level of presence during training (Vinayagamoorthy et al. 2004). Further, these cardiovascular measures can capture arousal and stress related to specific tasks such as aircraft takeoff and landing, time-pressured performance, and crisis-based activity (Cacioppo et al. 2000; Kramer 1991; Wilson 1992). However, when used in isolation, these measures have not always been found to be as sensitive as a workload index (Kaber et al. 2007, as cited in Stanney et al. 2009). The greatest utility of HRV within closed-loop systems is when it is combined with other physiological measures mainly because physical fitness of the individual, outside influences (e.g., drug use and nutrition), physical activity performed during cognitive evaluation, and daily circadian rhythms can substantially impact HRV measures. It is thus important to take each of these factors into account when incorporating HRV into cognitive sensor suites.

60.2.5 Pupillometry

Cognitive workload and emotional arousal are both known to be associated with pupil dilation. Using comparison of normalized means with temporal windows of 500-ms length, the difference in average pupil size from 0- to 500-ms postfixation onset and 2000- to 2500-ms postfixation onset has



(a) (b)

FIGURE 60.4 Equipment used to collect pupillometry data as part of Lockheed Martin Advanced Technology Laboratory's AugCog system. (a) Pupillometry display. (b) Head-mounted hardware to capture pupil changes.

been associated with an arousal-related effect (Partala and Surakka 2003), where the difference from 0 to 500 ms and 6000 to 6500 ms has been associated with a workload-related effect (Kahneman, Beatty, and Pollak 1967).

An Index of Cognitive Activity (ICA) developed under DARPA's AugCog program is based on binocular eye tracking sensors and measurement of pupil dilation (Kobus et al. 2005) (Figure 60.4). This system estimates cognitive activity from changes in pupil dilation (e.g., abrupt changes in pupil diameter indicate an operator's current levels of mental effort, whereas the point of gaze metric highlights specific elements causing difficulty [see Section 60.2.6]). Researchers found that it may be possible to determine different levels of utilization of left versus right brain hemisphere regions by comparing pupillometry data from each eye (Kobus et al. 2005). Such data could be used as an indicator of whether a user was using more verbal (left hemisphere) or spatial (right hemisphere) processing resources. In a more recent study, de Greef et al. (2009) found that pupil diameter significantly differed with workload; specifically, pupil diameter values collected during an underload scenario were significantly smaller that those collected during an overload scenario. However, the results failed to discriminate among all workload conditions (no significant difference was found between normal and overload scenarios). Conclusions from this study note that pupil diameter responds to many factors, with workload being one such factor. Further research is required to further advance pupillometry measures of cognitive state that are valid, reliable, and generalizable.

60.2.6 Eye/Gaze Tracking

Eye tracking technology offers a unique methodology for cognitive assessment in that systems can determine where visual attention is focused. By adding observation of the user's visual behavioral responses, insight can be gained into user's situation awareness (SA) of the ongoing situation and/or performance in completing the task at hand. Eye tracking has delivered promising results as a measure of cognitive load (Iqbal, Zheng, Bailey 2004), attention level (Fukuda and Yamada 1986), and task difficulty (Nakayama, Takahashi, and Shimizu 2002). Such oculomotor metrics that may be used to evaluate cognitive state in closed-loop systems include gaze direction and frequency, frequency/length of saccadic movements, pupil diameter, and eye blink. One study found that average fixation time significantly increased with increased mental workload (de Greef et al. 2009), yet found no significant differences in saccade distance or saccade speed. Similarly, King (2009) found increased fixations and longer fixation durations with increased task complexity.

One method for quantifying eye gaze data is to use the Nearest Neighbor Index (NNI), which is a ratio of the average nearest neighbor distances for fixation points (numerator) and the mean random distance (denominator) (Di Nocera et al. 2006). Preliminary results in one study suggested that NNI values may predict workload similar to NASA/TLX ratings (Fidopiastis et al. 2009). Further, eye fixations may be synchronized with other physiological indicators, such as EEG/ERP mentioned above, to create FLERPs, which can be used to evaluate cognitive states associated with specific events in defined visual locations (Hale et al. 2008).

60.2.7 Electromyograph

Skeletal muscle movements can be detected and measured via the electrical signals produced, using electromyograph (EMG) (Reaz, Hussain, and Mohd-Yasin 2006). EMG in psychological research is most often performed using electrodes on the skin surface to detect microtremors in muscle. When used in combination with other physiological measures (such as EEG), EMG can be used as an indicator of cognitive factors such as attention, effort, and stress (Harmon-Jones and Beer 2009). Along with EEG, ECG, pulse oxymetry, respiration, GSR, oculomotor, and facial temperature sensors, EMG has been integrated into aviation cockpit sensor suites (Schnell, Keller, and Macuda 2008).

60.2.8 Body Position/Posture Tracking

Body position and posture tracking can provide further supplemental information regarding cognitive state. Two such instantiations of these measures have been a pressure-sensing chair and a head tracker (Kobus et al. 2005). The sensor chair contains two 64-by-64 grids of pressure sensors in both the back and seat. Data from these sensors are used to compute changes in pressure over time, and the medial–lateral shifts of position in the seat posture shift data, head position, and head velocity, each output every 250 ms, were used to gauge levels of attention (e.g., engagement). Although a few experimental trials indicated ability to correlate with cognitive workload (Balaban et al. 2004), researchers determined more research would be needed to assess whether the body position/tracking system is reliably effective in predicting cognitive workload in command and control-type tasks.

60.2.9 Sensor Suites

In the last few years, AugCog researchers and developers have determined that various combinations of behavioral, physiological, and neurophysiological sensors may be used to improve the robustness for detecting cognitive functioning of users across a variety of conditions. Successful examples of integrated sensor suites are reviewed next. Each of the sensors/gauges discussed below was selected according to how effectively they assess cognitive state in real time, while also considering factors such as portability, usability in the field, potential intrusion on user task performance, and potential electromagnetic interference (EMI) during combined sensor/gauge implementation in operational settings.

A composite stress gauge (Kass et al. 2003; Raj et al. 2003) has been developed, which uses a weighted average of the following three sensor inputs to detect a participant's response to changes in cognitive load: Video pupillometry (VOG), high-frequency electrocardiogram (HFQRS ECG), and EDR. This gauge has been used to track the autonomic response to time pressure in a high-workload environment and to detect cognitive stress related to managing multiple competing tasks on a moment-to-moment basis. Thus, it is appropriate for assessing HIP bottlenecks related to attention (e.g., divided attention effects).

The Warfigher Interface Division of the U.S. Air Force Research Laboratory (AFRL) developed the new workload assessment monitor (NuWAM) sensor suite (Figure 60.5) based on an artificial neural net (ANN) cognitive state classifier of a 20-channel EEG system along with the ECG and electrooculography (EOG) sensors. This suite has been found to discriminate high from low workload levels for specific types of tasks that have been categorized during cognitive task analyses (Wilson and Russell 2006).

The cognitive cockpit (CogPit) sensor suite system's (Dickson 2005) primary function is to provide estimations of a pilot's cognitive-affective status in near real time while interacting in a fast-jet cockpit simulation environment (Figure 60.6). Pilot state inferences are derived from four main sensor sources (e.g., behavioral measures from interactions with cockpit controls, EEG-based physiological measures, subjective measures, and contextual information), which are used to assess the pilot's objective and subjective cognitive state in terms of arousal and workload levels (Pleydell-Pearce, Dickson, and Whitecross 2000). Cognitive state estimations are encapsulated within high-level state descriptors such as levels of stress, alertness, and workload and are then provided to a tasking interface manager (TIM) to aid in directing levels of automation or information presentation within the cockpit. CogPit can be configured to enable both online and post hoc operation.

More recently, there have been a number of systems that have integrated eye tracking and EEG. Merging high density EEG with unobtrusive eye tracking and head tracking measures as reported in Tucker and Luu (2009) allows for single-trial data measures and exact precision of timing of high-bandwidth data streams. A second such sensor suite uses eye tracking and EEG to capture FLERPs to assess the level of interest at specific visual fixation points utilizing neurophysiological indicators (Hale et al. 2007). Synching eye tracking and EEG has also proven effective at capturing

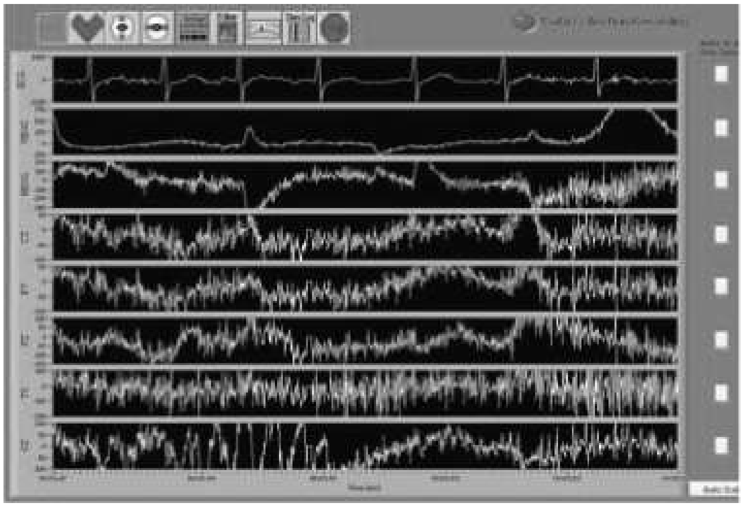


FIGURE 60.5 New workload assessment measure (NuWAM) sensor suite display.

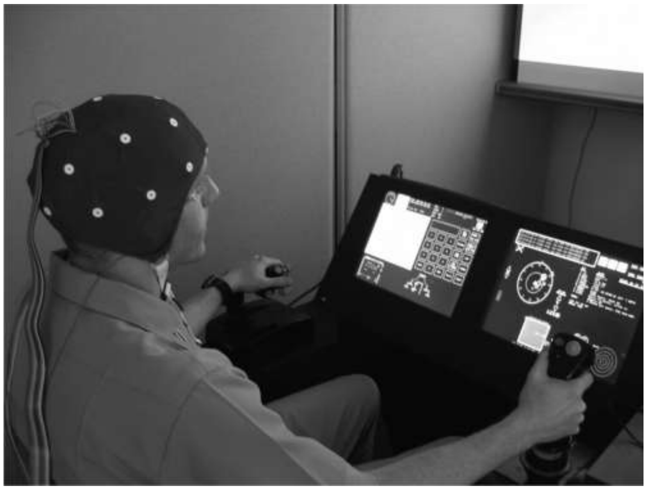


FIGURE 60.6 The cognitive cockpit.

snippets of interest from text data, where eye tracking is used to determine the timeline for EEG activity analysis (e.g., start and end of the sentence) (Hale et al. 2008). This sensor suite has more recently included heart rate measures and been integrated into an Auto-Diagnostic Adaptive Precision Training (ADAPT) framework developed to capture trainee state on a second-by-second basis, and evaluate trainees as they progress from novice to expert using a combination of behavioral events and overall cortical activity (Carroll et al. 2010).

60.3 DATA CLASSIFICATION AND INTEGRATION ARCHITECTURES

Given that AugCog systems will benefit most from a sensor suite of multiple gauges that together are used to assess operator state in real time, there is a need to ensure data classification and integration techniques utilize consistent methodologies to ensure gauges

- Relate to a cognitive state of interest (e.g., arousal and attention)
- Are reliable and accurate
- Are generalizable across applicable domain applications

This section outlines challenges related to real-time data analysis of multiple neurophysiological signals to derive cognitive state metrics that are practical for AugCog system integration.

60.3.1 Data Synchronization

One of the challenges in utilizing multiple sensor systems to evaluate cognitive state is data synchronization. This becomes of critical importance when using EEG/ERP analysis and synching this to eye fixations, as ERPs require synchronization at the millisecond level to pinpoint relevant event markers from the test bed or alternative sensor data (e.g. eye tracking). When using a Windows-based system, variable time delays occur between 1 and 40 ms, which are significant fluctuations for EEG/ERP analysis. Software solutions such as the Cognitive Avionics Tool Set have been developed to synchronize data input from numerous neurophysiological sensors (Schnell, Keller, and Macuda 2008).

60.3.2 State Classifiers

Under DARPA's AugCog program, there were four distinct cognitive state classifiers under development: sensory memory, working memory, executive function, and attention (Kobus et al. 2005). Since that time, researchers have expanded to create cognitive state indicators of interest (Kruse and Schulman 2006; Hale et al. 2007; Hale et al. 2008), decision making (Carroll et al. 2010), and expertise (where distinct EEG ERP components were evident as participants learned; Luu, Tucker, and Stripling 2007). Each of these state classifiers utilized a sensor suite that included neural, physiological, and behavioral measures to evaluate real-time status.

60.3.3 State-Based Architectures

Most early AugCog systems used a state-based architecture, where system mitigation was triggered based primarily on a cognitive state issue (e.g., overload). Although studies showed improvement in operator performance (Kobus et al. 2005), the type of mitigations instantiated were limited in scope, as there was little to no context understood in regards to *the root cause* of the cognitive state issue. For example, if sensory memory is overloaded, one mitigation strategy may be to hold back low-priority information and thus focus on presenting only the most time-critical information during this period. Although this mitigation can result in improved response times due to fewer information bits, the question of the longer term impact on higher level cognitive constructs, for instance SA, were not considered. In early systems, observed performance improvements (Barker et al. 2004; Dorneich et al. 2005; Whitlow and Ververs 2005) have "come at the cost of a loss of SA and survey knowledge of the environment" (Dorneich et al. 2004, p. iv).

60.3.4 Event-Based Architectures

Event-based cognitive assessment provides a more prescriptive way to evaluate cognitive processes in real time, by selecting key events as good/bad performance indicators, measuring the physiological reaction to these specific events, and configuring mitigation strategies "on the fly," depending on the combined system and operator state (Fuchs et al. 2008, 2007). Tracking and analysis of display events through an event-based approach allow for better context sensitivity, as cognitive gauges can now be related to individual events or even display objects, thereby providing the much needed context for how and where mitigations should be applied. In such an approach, each system event would be associated with increases or decreases in the cognitive state of interest (e.g., when a new entity appears on the screen, participants should perceive this change and act accordingly—this should result in increased attentional demands that impact workload). However, given that in complex high-workload systems (such as those in need of AugCog technology) there are usually numerous tasks occurring at any point in time, it is impossible to assume that every missed perceptual event would require mitigation. For example, events of low priority may be missed (i.e., not perceived) within a predefined time because the user's attention may currently be allocated to a higher priority task. In such a case, mitigation should not be triggered at this moment, as the operator is focused on a higher priority task. Disregarding this relationship may result in distraction from a critical task, as the mitigation strategy could, for example, try to shift the user's attention to a less critical event.

In event-based architectures, to ensure the priority of missed events compared to all active events is considered, each system event can be assigned a priority (based on a task analysis and/or SME input), and a dynamic list of active tasks in order of priority can be maintained in real time. In this way, mitigation would be triggered only if a high-priority event was missed. Once the mitigated event has been dealt with successfully (i.e., an appropriate behavioral response event has been detected), the event could be removed from the list and events with lower priority would move up. Missed lower priority events could then be flagged as potential mitigation candidates. Should lower priority events reach the top of the list, (i.e., higher priority tasks were successfully dealt with), mitigations could be triggered for these events, as well. Thus, to regulate mitigation in an event-based architecture, a missed perceptual event could be a first criterion, followed by the priority of the missed event.

60.4 REAL-TIME MITIGATION

Although there have been a number of innovative mitigation strategies identified for applicability within AugCog systems (Fuchs et al. 2007), AugCog systems to date have been limited in the number and variety of real-time mitigation strategies, in part due to limited knowledge regarding context (Fuchs et al. 2008). Thus, many prior implementations of AugCog systems had a problem with inconsiderate augmentation (a term coined by Stanney and Reeves 2005), where SA was sometimes compromised for performance benefits. The below sections (60.4.1, 60.4.2, and 60.4.3) outline various mitigation strategies that may be implemented to enhance performance in an AugCog system. By definition, an AugCog mitigation strategy is "an intervention technique that is triggered by the outcome of cognitive state assessment and context to significantly improve human–system performance" (Reeves et al. 2007b, p. 282). Such a system manipulation has associated benefits and costs, and should be implemented when the benefits to performance outweigh the associated costs. Various strategies have been proposed to adapt what information is presented and how information presentation should change in real time based on OFS to optimize human performance. To address what should be presented, the architectures discussed above (state-based and event-based) should be utilized. As mentioned above, although state-based architectures in isolation may effectively guide when adaptation is needed, they may not provide enough detail about the ongoing situation to effectively guide what information should be mitigated and how it should be presented (e.g., what information should be adapted and for how long).

60.4.1 Adaptation of the Presentation

One group of mitigation strategies involve changing how information is presented. This may include strategies such as modality augmentation and modification of information type (e.g., verbal or spatial) (Fuchs et al. 2007). Modality augmentation includes both switching one sensory modality with another to optimize distribution of processing load and redundancy, which provides complementary information in a second modality. This can also be thought of as "cueing," where salience of information is increased by adding redundant cues (Fuchs et al. 2008). Switching is designed to effectively reduce overload in a given sensory channel, whereas redundancy can improve performance through enhanced clarification of information. Changing the information type is referred to as transposition, where spatial information may be switched to verbal information, for example, to optimize information processing, as suggested by the Multiple Resource Theory (Wickens 2002). Multiple empirical studies (Diethe 2005; Dorneich et al. 2006) have shown that humans can effectively process multiple bits of information from differing resource pools (e.g., spoken directions as a verbal transposition of a visual-spatial map), and that distributing information across available resources improves performance.

Another form of presentation adaptation includes decluttering, where the amount and/or level of detail of information is reduced with the goal of making remaining information more salient. This may include decreasing saliency of lower priority information, reducing the level of detail, or removing information completely to decrease the complexity of presented information (Fuchs et al. 2008).

60.4.2 Adaptation of the Schedule

Real-time mitigation can adjust the timing associated with information presentation, such that information is paced or sequenced in such a way as to optimize performance. Pacing refers to holding back low-priority information until current high-priority tasks are complete (Tremoulet et al. 2005). The goal of such a strategy is to ensure high-priority information that is addressed in a timely manner by reducing the amount of distracting information presented, thereby allowing focus on critical tasks. Sequencing involves the decomposition of tasks into chunks, where multiple tasks may be time shared through rearrangement of task chunks to optimize information processing (Fuchs et al. 2008) while minimizing effects of task switching and task interruption.

60.4.3 Adaptation of System Autonomy

System autonomy is designed to reduce the human's cognitive load by offloading certain tasks or subtasks to the software system. Two methods that have been utilized in AugCog systems include context-sensitive help and mixed initiative strategies (Fuchs et al. 2008). Context-sensitive help inserts task-specific information at the time help is needed (Sukaviriya and Foley 1990), thereby minimizing the need to search for assistance or the risk of missing information that a human might otherwise overlook (Kirsh 2000). Mixed initiative (or adaptive automation) (Scerbo 2001) adjusts the level of control between the human and system dynamically to optimize human performance. This switching can be driven by operator cognitive state alone or by a combination of metrics that provide insights into OFS, system state, and task state to ensure automated systems are effectively assisting in optimizing performance and not causing operator cognitive challenges such as task interruption (by offloading a current task midstream) or loss of SA.

60.5 APPLICATION OF NEUROTECHNOLOGIES

Although other application areas may emerge in the near future, the most appropriate application areas that may presently benefit from effective implementation of current neurotechnologies include: (1) operational systems, (2) training systems, (3) operator selection, and (4) HSI design of systems. This section reviews some of the S&T developments in these areas and implications emerging from the AugCog community.

60.5.1 Operational Systems

Leveraging physiological and neurophysiological technologies, it is now possible to assess the ever-changing cognitive state of the user in real time and mitigate against human performance limitations caused by known HIP bottlenecks or other nonoptimal cognitive states (e.g., disengagement, distraction, and drowsiness). Essential elements of any closed-loop AugCog system must therefore include the following:

- Operator functional assessment capabilities via physio-and neurophysiological tools as discussed in Section 60.2.
- Methods for classifying the data from these tools to identify periods of nonoptimal state (e.g., performance and cognitive state) to drive real-time system mitigation via adaptive automation techniques (e.g., AugCog mitigation strategies).
- An integration architecture that synchronizes and optimizes all necessary components via a robust controller (Reeves et al. 2007b) or mitigation management architecture that maintains system stability (e.g., controlling the "when," "what," and "how" of the adaptive automation techniques).

Under DARPA's AugCog and IWIIUS programs, a number of operational environments and subsequent prototype systems were developed to demonstrate the benefit of a real-time, closed-loop system that utilizes neurophysiological indicators of operator cognitive state to drive optimized human performance. One application domain included the U.S. Army's future force warrior (FFW) program (Kobus et al. 2005) where the prototype AugCog system demonstrated a 380% performance improvement where attention resources were required. In addition, the system was able to correctly classify attention state changes more than 98% of the time and in less than 300 ms. A second domain was light-armored vehicles, where the prototype AugCog system under real-world driving conditions showed that the sensory bottlenecks could be improved by as much as 108% with an accuracy of up to 98%, depending on the modality being examined. The sensory bottleneck status could be detected in as little as 200 ms, and mitigations could be invoked in as little as 0.2 sec, depending on the mitigation being used. A third domain was the Tactical Tomahawk Weapons Control System (TTWCS) simulation environment (Kobus et al. 2005), which is a command and control station that requires management of a number of missiles, targets and shipboard launch platforms, and dynamic reassignment of missiles to targets as critical targets pop-up, missiles fail, and so forth. The prototype AugCog system in this domain demonstrated an improvement in working memory throughput by at least 500% by using an intelligent sequencing mitigation strategy to strategically present related information about specific missile-target pairings when the working memory bottleneck was saturated. Working memory status (high- or low-cognitive workload) was correctly identified in over 90% of the trials, with the sequencing mitigation being invoked in less than 500 ms. A fourth domain was unmanned air vehicle (UAV) systems, where the prototype AugCog system showed a 241% performance improvement in executive function-related tasks, with a classification accuracy of 92% in less than 1 sec.

60.5.1.1 Training Systems

Various noninvasive brain monitoring technologies have been successfully applied to the challenge of documenting the transformation from novice to expert in a variety of domains, including the following: identifying indices of skill level in basic laboratory tasks and marksmanship (Ciesielski and French 1989; Deeny et al. 2003; Kerick, Douglass, and Hatfield 2004); identifying indices of skill acquisition in computer games (Smith, McEvoy, and Gevins 1999), and detection of the progression toward automaticity of syntactic processing (Gunter and Friederici 1999). A recent study examining marksmanship skills found significant differences in EEG parameters between novices and experts (Pojman et al. 2009). In addition, experts were found to have lower HRV scores compared to novices during the least cognitively challenging task. Authors note that "it is unclear whether this apparent ability to regulate expert's physiology is a genetically determined trait or a skill that can be acquired and refined with training" (Pojman et al. 2009, p. 531). Additional studies have also documented brain activity patterns that are indicative of a shift in cognitive processing from focused conscious effort to "automatic" processing (Floyer-Lea and Matthews 2004; Peres et al. 2000). This shift to automatic processing may occur after task performance has leveled off and is significant because it marks the point at which minimal cognitive effort is required to perform the task well. This transition may serve as an indication of the point at which the trainee is ready to learn additional components of the task or be taught higher level strategies within the particular domain.

The identification of neurophysiological measures that correlate with task and skill mastery is of value for three practical reasons:

1. It could serve as the basis for real-time, closed-loop training systems that adapt to the individual's current state of task proficiency and automaticity; this application would have the most profound impact on the training capabilities, but is not ready for widespread distribution.
2. Prior to widespread distribution for closed-loop training, AugCog-based technologies could be used to evaluate alternative training strategies and systems to identify those that provide the most effective learning environment. This application is potentially feasible in the near future, but would require more upfront research to confirm the neurophysiological patterns associated with task mastery are applicable to more complex task environments.
3. During the course of this type of research, novel training strategies may emerge that are validated by laboratory observations of accelerated task mastery and automaticity, but which do not rely on real-time monitoring of brain activity to enact. Such observations would potentially be immediately transferable to instructional environments.

One example of an AugCog training system that has been developed is the Quality of Training Effectiveness Assessment (QTEA) tool (Schnell et al. 2009). This system builds on the measurement capabilities outlined above and quantifies the student's workload level in real time to drive scenario manipulations. In addition, the cognitive and physiological measures also serve as "a quantitative manifestation of a student's learning curve" (Schnell et al. 2009, p. 641). A second AugCog system showed that workload and engagement levels through training were correlated with eventual performance outcomes, and thus, provided evidence that cognitive state gauges can provide predictive assessment of training outcome prior to completion of the training (Craven et al. 2009). Developing such AugCog-based training systems could have substantial impact on the training community, including the following:

- Enhanced instructor understanding of a trainee's capabilities and limitations
    - More suitable operator assignments to match skill levels
    - Ability to predict fast learners early during training
- More timely delivery of information provided to trainee
- Increased learning capabilities (e.g., via an increase of an individual's cognitive processing capabilities during training)
- More expertly trained and mentally prepared operators
- Enhanced training acquisition
- More effective training per unit time (reduce required financial investment)
- Enhanced development of effective individual and team performance (possibly in real time)

60.5.2 Operator Selection

AugCog systems developed for assessment of training progression could also be used to determine an individual's cognitive potential in general or their potential to reach a certain skill level in a particular task domain. For instance, the military has been using screening tools for decades to select for fighter pilots before thousands of dollars and man hours are wasted in training someone who may never be cognitively equipped to attain the necessary skill level required for such a domain. AugCog enabled screening tools could be used for similar purposes to improve the diagnostic capabilities of existing screening strategies.

Assessment of capabilities via an extensive neurophysiological and neuropsychological/psychometric components skills battery would be required in such an effort. An example of such an approach could be to determine the neural correlates of the knowledge, skills, and attitudes (and/or abilities) (KSAs) required for a particular task domain and training level. Individuals with the greatest human performance results related to the required KSAs for the particular task domain could be used as the expert model with which to compare other individual's neurophysiological indicators and patterns. This approach would also need to leverage the novice-to-expert training progression assessment techniques discussed above to establish the KSA neurolevel correlates along the entire training progression continuum for necessary comprehensive comparative evaluations. Individuals could be assessed initially to determine a priori how suited they may or may not be for a particular task domain (e.g., aptitude assessment). They could later be assessed to determine how their training is progressing at both the human performance and neurolevels. The KSA approach is only a recommendation, as other neurolevel correlates of performance and aptitude may also be used. Specific benefits of any neurolevel screening/selection approach could include the following:

- Identifying which individuals may be most apt to reach deploy-ready or expert levels for a given task domain and/or who may never have the "aptitude" to get to required levels
- Identifying operators with the greatest capacities for information processing and decision making, and then assigning them to critical information operations
- Identifying users who could be trained to deployable-or expert-level more quickly than others
- Reassessing trained individuals to see when retraining may be necessary (e.g., to reacquire necessary KSA levels)

60.5.3 Improving HCI System Design and Evaluation Capabilities

The ability to build systems truly tailored to a user's HIP needs and capabilities is now possible. Based on the review of AugCog technologies presented in Sections 60.2, 60.3, and 60.4, it is evident that HCI practitioners now have available to them various physiological- and neurophysiological-based tools and techniques that may be used to better specify both the design and evaluation of human–computer systems. Craven et al. (2009) propose using AugCog cognitive state indicators to drive the type of training provided to ensure the overall training program optimizes workload and engagement while avoiding distraction and drowsiness states. For example, "Including mental state gauges as part of a computer-based training solution would allow the system to make the individual adjustments necessary at distinct points in the training without waiting until the training has concluded and poor posttest performance indicates that they likely lost focus at some point during training" (Craven et al. 2009, p. 594).

Most any of the neurophysiological tools and techniques implemented in the AugCog community could be used and applied to the design and evaluation of most any human–computer system, resulting in more rapid, effective, and eventually less expensive HCI design and evaluation processes.

60.6 LESSONS LEARNED AND FUTURE DIRECTIONS

Over the last decade, AugCog technology has seen rapid advances in real-time cognitive state detection and applications of closed-loop systems in operational and training environments. Despite the great advances achieved to date, there are a number of lessons learned for AugCog system design. Below is a list of such lessons learned from the last decade of AugCog system research and development.

- The need to understand the user. A cognitive task analysis should be conducted in order to fully understand the cognitive requirements of the operator. Having a clear understanding of who the user is, as well as thoroughly understanding what they do, is essential for successfully constructing future AugCog technologies.
- Integrate early, integrate often. The earlier and more thorough system integration (e.g., of the various sensors, computer hardware and software) was considered, the fewer the setbacks and more effective the mitigations. Slight modifications in sensors should not be treated superficially and require systematic assessment throughout development.
- One sensor/gauge does not fit all. The selection of specific gauges for use within the AugCog systems is highly context (task) dependent. Gauges that are well suited for one environment may or may not be applicable to another domain.
- Cognitive bottlenecks may be task specific. Although the original DARPA AugCog program focused specifically on distinct cognitive bottlenecks, it appears that when applying these concepts to real-world tasks, they are often overlapping or interacting in nature. More recent research and development efforts have found it useful to define bottlenecks and gauges in terms of the tasks that are to be mitigated, vice attempting to address conceptual bottlenecks that are not adequately defined in terms of the operational task. It should be noted, however, that regardless of the operational definition of a specific bottleneck, the net result of the mitigation strategies employed by AugCog teams improved overall task performance.
- Artifact detection and correction. A continuing challenge to the AugCog community is the ability to differentiate between physiological changes related to cognitive activity vice the physical requirements of the task. Such an issue may be even more of a concern for applications involving mobile users. The critical need is to be able to reduce the effect of the motion artifacts in order to continue to provide adequate estimations of cognitive state.
- Perceived stability/trust. An issue with stability and predictability of AugCog mitigation strategies concerns the use of a combination of physiologically based gauges in conjunction with context-based sensors. If an operator does not understand the logic (e.g., the rationale for triggering the onset/offset of the mitigation), there is a significant potential for disruption and degraded performance. Another issue relates to the impact of degraded sensors. If users are not aware of the degraded functionality of the system, their expectations of augmentation

would not be met, and the system may be perceived as unstable. Before such a system is deployed, training strategies will need to be developed to allow users to learn how to operate in the face of degraded augmentation.

- Timing of mitigation strategies. Many of the changes in perceived cognitive load may be momentary. The physiological measures being used as gauges may detect either a transient or sustained change in cognitive activity. It may be unrealistic, or at least have little practical application, to have a system that is sensitive enough to detect and implement mitigations in terms of seconds. It is clear that in the operational environment, momentary changes in one or more of the cognitive bottlenecks areas may occur rapidly (1 min). More research needs to be conducted to investigate optimal timing for implementing various mitigation strategies, including on/off strategies of when and how to transition in and out of mitigated states.
- Hardware integration. As sensor technologies improve in their sensitivity and robustness, it is anticipated that calibration process will be more feasible for field conditions, as it is critical that AugCog technologies can be easily donned and readily calibrated. Significant consideration must be given to potential interference sources in the field (particularly when wireless devices are used), and potential design solutions that could avoid such issues in the first place. Integrated sensors designed to be used in the field from the outset are more likely to adequately address this issue than technologies adapted from laboratory or medical applications. System integration architectures will need to be refined and streamlined in order to improve new sensor integration and to reduce the processing demands on any emerging sensor integration requirements.
- Individual differences. Past AugCog research has noted challenges with both interindividual and intraindividual differences, particularly for the EEG-derived gauges. Another aspect of the underlying EEG phenomena that is poorly understood is the effect of extensive experience with the tasks on the utility of EEG-derived gauges. It appears that successful utilization of EEG data in an AugCog system may require that the filters used in separating signals from artifacts will need to be tailored to every user and even be dynamically adaptable during the course of use of the AugCog system.
- Proactive vice reactive AugCog. AugCog closed-loop systems developed to date are largely reactive in that the user must first become overloaded (or at least close to overloaded) before the system will invoke a mitigation strategy. Although it is important to detect when such levels of activity occur, it would be of great operational importance to develop systems that are also predictive (proactive) rather than simply reactive.

In addition to lessons learned, there are a number of continuing challenges that AugCog system designers face. Below is a list of future research and development directions organized by key area that will be key to driving the community of AugCog forward in developing revolutionary closed-loop systems that optimize human performance.

- Cognitive-state sensors
  - Designing future sensors for ease of use, calibration, appropriate resolution/sensitivity, noise cancellation, less invasiveness, and accommodation of individual user variability
  - Designing future gauge algorithms that accommodate day-to-day fluctuations and skill acquisition
  - Enhanced understanding of neurological, psychological, and cognitive theories that should be driving sensor placement, data analysis, and subsequent "cognitive load" and/or "cognitive state" gauge derivation
  - Determining appropriate experimental techniques in applied task settings to assess effectiveness of sensors to accommodate both general use settings and individual differences across task domains
- Mitigation strategies
  - Pursue only those mitigation strategies that affirm the goals of AugCog (e.g., those that extend, by an order of magnitude or more, the information management capacity of the human–computer integral).
  - Seek only to implement mitigation strategies based on objective and scientifically valid human performance assessment.
  - Pursue only those mitigation strategies that are operationally feasible (either now or in the foreseeable future).
  - Identify how/when individual differences will affect appropriate mitigation strategy choices and determine how to manage these differences such that human performance is enhanced for all.
  - Enhance the effectiveness of mitigation strategy implementation by identifying which strategies may be user, context, or domain dependent and which are generalizable across these dimensions.
  - Validate entry/exit and transition techniques to ensure optimal system mitigation, where mitigations are applied at opportune moments and transitioned out of at the earliest possible moment.
  - Leverage the arts to develop truly innovative mitigation strategies that are "invisible" to the user, particularly as AugCog S&T is applied to new display devices (e.g., PDAs, augmented reality systems, etc.).
  - Consider how mitigation strategies may impact a team training environment and develop validated approaches to optimize team training and operations.

- Robust controllers
  - Given the complexit.y of HIP, mathematical approaches are needed, but are not the sole answer to developing robust controllers, as there is a need for real-time user and system models based on complex human biology and physiology, as well as a need for task models, which take all contexts into account.
  - More basic and applied research is needed to develop sufficiently comprehensive and accurate models of the components (e.g., user, task, and system contexts) that are to be controlled, as well as determining how to control them with appropriate approaches, be they mathematical or otherwise.
  - Any robust controller should be stable and seamless, where a user is unaware of when a controller is being used yet trusts the system when subsequent effects are noticed (e.g., mitigation turned on/off) and therefore benefits and not suffers from the effects.
  - Both users and funding sponsors need proof of the effectiveness of any controller's ability to integrate the human within the system-of-systems architecture, where input and output of information flow and mitigation control are sufficiently adaptable to improve user performance and maintain overall system stability.
  - Return on investment (ROI) must be justified in terms of development cost (time and money) and benefits (significant user/system performance improvements).
- Roles of National and Supranational Institutions
  - Continued need for more funding to be funneled into the development of AugCog sensors from national and supranational institutions (e.g., DARPA, ONR, NIH, and NSF).
  - Institutions need to foster the ability of HCI practitioners to begin identifying uses for neurotechnologies in various application domains.

60.7 CONCLUSIONS

The field of AugCog has emerged in large part as a result of substantial investment from DoD-funded programs and projects that have been focused on developing neurophysiological-based tools and techniques to enable revolutionary changes in HCI system design. Whether the application context is an operational closed-loop system, system design and evaluation, or education and training, such tools and techniques offer the ability to create human–machine synergy and optimization never before realized. Similar to Licklider's original visions of human–machine symbiosis in 1960, AugCog researchers and practitioners aim to build tightly coupled brain–machine interfaces that surpass the information-handling capacity of traditional HCI systems and empower one operator with the ability to perform a job normally required of two or more operators. Such an improvement in the human–computer integral is a worthy goal. Being able to noninvasively measure and assess users' cognitive state in real time, and then use automated computational systems to modify and enhance HIP capabilities of these users in any application context, is a goal that could substantially improve human performance and the way humans interact with computer-based systems in the twenty-first century. It is up to HCI researchers and practitioners to begin implementing various AugCog tools and techniques by selecting the most appropriate neurophysiological tools for their task applications and users. The technologies reviewed in this chapter will hopefully provide a nice "initial" tool set from which to begin such a selection process.

REFERENCES

Barker, R. A., R. E. Edwards, K. R. O'Neill, and R. J. Tollar. 2004. *DARPA Improving Warfighter Information Intake Under Stress—Augmented Cognition Concept Validation Experiment (CVE) Analysis Report for the Boeing Team (Contract NBCH030031).* Arlington, VA: Defense Advanced Research Projects Agency.

Berka, C., D. J. Levendowski, M. N. Lumicao, A. Yau, G. Davis, V. T. Zivkovic, R. E. Olmstead, P. D. Tremoulet, and P. L. Craven. 2007. EEG corrleates of task engagement and mental workload in vigilance, learning, and memory tasks. *Aviat Space Environ Med* 78(5):B231–44.

Bradley, M. M., B. Moulder, and J. P. Lang. 2005. When good things go bad: The reflex physiology of defense. *Psychol Sci* 16:468–73.

Cacioppo, J. T., G. G. Berntson, J. F. Sheridan, and M. K. McClintock. 2000. Multilevel integrative analyses of human behavior: Social neuroscience and the complementing nature of social and biological approaches. *Psychol Bull* 126:829–43.

Carroll, M., S. Fuchs, A. Carpenter, K. Hale, R. G. Abbott, and A. Bolton. 2010. Development of an autodiagnostic adaptive precision trainer for decision making (ADAPT-DM). *Int Test Eval J* 31(2):247–263.

Ciesielski, K. T., and C. N. French. 1989. Event-related potentials before and after training: Chronometry and lateralization of visual N1 and N2. *Biol Psychol* 28(3):227–38.

Craven, P. L., P. D. Tremoulet, J. H. Barton, S. J. Tourville, and Y. Dahan-Marks. 2009. Evaluating training with cognitive state sensing technology. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 585–94. Springer-Verlag Berlin Heidelberg: Germany.

Critchley, H. D. 2002. Electrodermal responses: What happens in the brain. *Neuroscientist* 8(2):132–42.

Deeny, S. P., C. H. Hillman, C. M. Janelle, and B. D. Hatfield. 2003. Cortico-cortical communication and superior performance in skilled marksmen: An EEG coherence analysis. *J Sport Exerc Psychol* 25:188–204.

de Greef, T., H. Lafeber, H. van Oostendorp, and J. Lindenberg. 2009. Eye movement as indicators of mental workload to trigger adaptive automation. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 219–28. Springer-Verlag Berlin Heidelberg: Germany.

Dickson, B. T. 2005. The cognitive cockpit—A testbed for augmented cognition. In *CD Proceedings of the First International Conference on Augmented Cognition*, 11:479–88. Las Vegas, NV: Nevada.

Diethe, T. 2005. The future of augmentation managers. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, 631–40. Mahwah, NJ: Erlbaum.

Di Nocera, F., M. Terenzi and M. Camilli. 2006. Another look at scanpath: distance to nearest neighbor as a measure of mental workload. In D. de Waard, K. Brookhuis and A. Toffetti. *Developments in Human Factors in Transportation, Design, and Evaluation* (pp. 1–9). Maastricht, the Netherlands: Shaker Publishing.

Dorneich, M. C., P. M. Ververs, S. Mathan, and D. S. Whitlow. 2006. Evaluation of a tactile navigation cueing system triggered by a real-time assessment of cognitive state. In *Proceedings of the Human Factors and Ergonomics Society 50th Annual Meeting*, 2600–4. Santa Monica, CA: Human Factors and Ergonomics Society.

Dorneich, M. C., S. D. Whitlow, S. Mathan, J. Carciofini, and M. P. Ververs. 2005. The communications scheduler: A task scheduling mitigation for a closed loop adaptive system. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, 132–41. Mahwah, NJ: Erlbaum.

Dorneich, M., S. Whitlow, P. M. Ververs, J. Carciofini, and J. Creaser. 2005. Closing the loop of an adaptive system with cognitive state. In *Proceedings of the 48th Annual Meeting of the Human Factors and Ergonomics Society, Santa Monica, CA: Human Factors and Ergonomics Society.* pp. 590–594.

DuRousseau, D. R. 2004. *Multimodal Cognitive Assessment System (Final Tech. Rep. DAAH01-03-C-R232).* Arlington, VA: Defense Advanced Research Projects Agency.

Duta, M., C. Alford, S. Wilon, and L. Tarassenko. 2004. Neural network analysis of the mastoid EEG for the assessment of vigilance. *Int J Hum Comput Interact* 17(2):171–95.

Fidopiastis, C. M., J. Drexler, D. Barber, K. Cosenzo, M. Barnes, J. Y. C. Chen, and D. Nicholson. 2009. Impact of automation and task load on unmanned system operator's eye movement patterns. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 229–38. Springer-Verlag Berlin Heidelberg: Germany.

Floyer-Lea, A., and P. M. Matthews. 2004. Changing brain networks for visuomotor control with increased movement automaticity. *J Neurophysiol* 92(4):2405–12.

Freeman, F. G., P. J. Mikulka, L. J. Prinzel, and W. M. Scerbo. 1999. Evaluation of an adaptive automation system using three EEG indices with a visual tracking system. *Biol Psychol* 50:61–76.

Fuchs, S., K. S. Hale, C. Berka, and J. Juhnke. 2008. Enhancing situation awareness with an augmented cognition system. In *Augmented Cognition: A Practitioner's Guide*, ed. K. M. Stanney and D. Schmorrow, 112–43. Santa Monica, CA: HFES.

Fuchs, S., K. S. Hale, K. M. Stanney, J. Juhnke, and D. Schmorrow. 2007. Enhancing mitigation in augmented cognition. *J Cogn Eng Decis Mak* 1:309–26.

Fukuda, T. and M. Yamada. 1986. Quantitative evaluation of eye movements as judged by sight-line displacements. *SMPTE* Mot. Imag *J* (95)12:1230–41.

Gratton, G., A. F. Kramer, and M. Fabiani. 2008. Brain sensors and measures. In *Augmented Cognition: A Practitioner's Guide*, ed. D. D. Schmorrow and K. M. Stanney, 1–26. Santa Monica, CA: Human Factors and Ergonomics Society.

Gunter, T. C., and A. D. Friederici. 1999. Concerning the automaticity of syntactic processing. *Psychophysiology* 36(1):126–37.

Hale, K. S., S. Fuchs, P. Axelsson, A. Baskin, and D. Jones. 2007. Determining gaze parameters to guide EEG/ERP evaluation of imagery analysis. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, D. M. Nicholson, J. M. Drexler and L. M. Reeves, 4th ed., 33–40. Arlington, VA: Strategic Analysis Inc.

Hale, K. S., S. Fuchs, P. Axelsson, C. Berka, and A. J. Cowell. 2008. Using physiological measures to discriminate signal detection outcome during imagery analysis. *Proc Hum Fact Ergon Soc Annu Meet* 52(3):182–86.

Harmon-Jones, E., and S. J. Beer. 2009. *Methods in Social Neuroscience*. New York: Guilford Press.

Healey, J. A. 2000. *Wearable and Automotive System for Affect Recognition from Physiology*. Unpublished Ph.D. thesis, Massachusetts Institute of Technology. Cambridge, MA.

Hoover, A., and E. Muth. 2004. A real-time index of vagal activity. *Int J Hum Comput Interact* 17(2):197–210.

Iqbal, S. T., X. S. Zheng, and B. P. Bailey. 2004. Task-evoked pupillary response to mental workload in human-computer interaction. *Proc ACM Conf Hum Fact Comput Syst* 1477–80. http://interruptions.net/literature/Iqbal-CHI04-p1477-iqbal.pdf

Izzetoglu, K., S. Bunce, B. Onaral, K. Pourrezaei, and B. Chance. 2003. Functional optical brain imaging using near-infrared during cognitive tasks. *Int J Hum Comput Interact* 17(2): 211–27.

Izzetoglu, M., K. Izzetoglu, S. Bunce, H. Ayaz, A. Devaraj, and B. Onaral, et al. 2005. Functional near-infrared neuroimaging. *IEEE Trans Neural Syst Rehabil Eng* 13:153–59.

Jang, D. P., I. Y. Kim, S. W. Nam, B. K. Wiederhold, M. D. Wiederhold, and I. S. Kim. 2002. Analysis of physiological response to two virtual environments: Driving and flying simulation. *Cyberpsychol Behav* 5(1):11–8.

Jung, T. -P., S. Makeig, M. Stensmo, and T. J. Sejnowski. 1997. Estimating alertness from the EEG power spectrum. *IEEE Transactions on Biomedical Engineering* 44(1):60–9.

Kaber, D. B., C. M. Perry, N. Segall, and M. A. Sheik-Nainar. 2007. Workload state classification with automation during simulated air traffic control. *Int Aviat Psychol* 17:371–90.

Kahneman, D., J. Beatty, and I. Pollack. 1967. Perceptual deficit during a mental task. *Science* 157:218–19.

Kass, S. J., M. Doyle, A. K. Raj, F. Andrasik, and J. Higgins. 2003. Intelligent adaptive automation for safer work environments. In *Occupational Health and Safety: Encompassing Personality, Emotion, Teams, and Automation*, ed. J. C. Wallace and G. Chen (Co-Chairs), Symposium conducted at the Society for Industrial and Organizational Psychology 18th Annual Conference, April. Orlando, FL.

Kerick, S. E., L. W. Douglass, and B. D. Hatfield. 2004. Cerebral cortical adaptations associated with visuomotor practice. *Med Sci Sports Exerc* 36:118–29.

King, L. A. 2009. Visual naviagation patters and cognitive load. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 254–59. Springer-Verlag Berlin Heidelberg: Germany.

Kirsh, D. 2000. A few thoughts on cognitive overload. *Intellectica* 30:19–51.

Kobus, D. A., C. M. Brown, J. G. Morrison, G. Kollmorgen, and R. Cornwall. 2005. *DARPA Improving Warfighter Information Intake under Stress—Augmented Cognition Phase II: The Concept Validation Experiment (CVE)*. DARPA/IPTO technical report submitted to CDR Dylan Schmorrow.

Kollmorgen, L. S. 2007. Introduction: A case for operational approach in advanced research projects—the augmented cognition story. *Aviat Space Environ Med (Spec Suppl)* 78(5), B1–B3.

Kramer, A. F. 1991. Physiological metrics of mental workload: A review of recent progress. In *Multiple Task Performance*, ed. D. L. Damos, 279–328. Washington, DC: Taylor & Francis.

Kruse, A. A., and J. J. Schulman. 2006. Neurotechnology for Intelligence Analysts. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, K. M. Stanney, and L. M. Reeves, 2nd ed., 27–31. Arlington, VA: Strategic Analysis, Inc.

Lee, C. K., S. Yoo, Y. J. Park, N. Kim, K. Jeong, and B. Lee. 2005. Using neural network to recognize human emotions from heart rate variability and skin resistance. In *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 5523–25. Los Alamitos, CA: IEEE Computer Society.

Licklider, J. C. R. 1960. Man-computer symbiosis. *IRE Trans Hum Factors Electron* 1:4–11.

Luu, D., D. M. Tucker, and R. Stripling. 2007. Neural mechanisms for learning actions in context. *Brain Res* 1179:89–105.

Mandryk, R. L. 2005. *Modeling User Emotion in Interactive Play Environments: A Fuzzy Physiological Approach*. Unpublished Ph.D. thesis, Simon Fraser University. Burnaby, BC.

Mathan, S., and S. Whitlow, et al. 2006. Neurophysiologically driven image triage: a pilot study. *Conference on Human Factors in Computing Systems, Montréal, Québec*. Canada ACM Press New York, NY.

Morrison, J. G., D. A. Kobus, and M. C. Brown. 2006. *Volume I: DARPA Improving Warfighter Information Intake under Stress—Augmented Cognition*. Phase II Concept Validation (Tech. Rep. 1940). San Diego, CA: Pacific Science and Engineering Group. http://handle.dtic.mil/100.2/ADA457526 (accessed June 30, 2009).

Nakayama, M., K. Takahashi, and Y. Shimizu. 2002. The Act of task difficulty and eye-movement frequency for the 'oculo-motor indices.' *Proc Symp Eye Track Research Appl* 37–42.

Parra, L. C., C. D. Spence, A. D. Gerson, and P. Sajda. 2005. Recipes for the linear analysis of EEG. *Neuroimage* 28:326–41.

Partala, T. and V. Surakka. 2003. Pupil size variation as an indication of affective processing. *Int J Hum Comput Stud* 59:185–98.

Peres, M., P. F. Van De Moortele, C. Pierard, S. Lehericy, P. Satabin, and D. Le Bihan, et al. 2000. Functional magnetic resonance imaging of mental strategy in a simulated aviation performance task. *J Aviat Space Environ Med* 71(12):1218–31.

Pleydell-Pearce, K., B. Dickson, and S. Whitecross. 2000. Cognition monitor: A system for real time pilot state assessment. In *Contemporary Ergonomics 2000*, ed. P. T. McCabe, M. A. Hanson, and S. A. Robertson, 65–9. London: Taylor & Francis.

Pleydell-Pearce, C. W., S. E. Whitecross, and B. T. Dickson. 2003. Multivariate analysis of EEG: Predicting cognition on the basis of frequency decomposition, inter-electrode correlation, coherence, cross phase, and cross power. In *Abstract Proceedings of the 36th Annual Hawaii International Conference on System Sciences (Track 5, CD-ROM)*. Los Alamitos, CA: IEEE Computer Society.

Pojman, N., A. Behneman, N. Kintz, R. Johnson, G. Chung, S. M. Nagashima, P. Espinosa, and C. Berka. 2009. Characterizing the psychophysiological provile of expert and novice marksmen. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 524–32. Springer-Verlag Berlin Heidelberg: Germany.

Pope, A. T., E. H. Bogart, and D. S. Bartolome. 1995. Biocybernetic-system validates index of operator engagement in automated task. *Biol Psychol* 40:187–95.

Prinzel III, L. J., F. C. Freeman, M. W. Scerbo, P. J. Mikulka, and A. T. Pope. 2000. A closed-loop system for examining psychophysiological measures for adaptive task allocation. *Int J Aviat Psychol* 10:393–410.

Raj, A. K., J. F. Perry, L. J. Abraham, and A. H. Rupert. 2003. Tactile interfaces for decision making support under high workload conditions. In *Proceedings of Aerospace Medical Association 74th Annual Scientific Meeting*. San Antonio, TX.

Reaz, M. B. I., M. S. Hussain, and F. Mohd-Yasin. 2006. Techniques of EMG signal analysis: Detection, processing, classification and applications. *Biol Proced Online* 1(8):11–35.

Reeves, L. M., D. D. Schmorrow, and K. M. Stanney. 2007a. Augmented cognition and cognitive state assessment technology–near-term, mid-term, and long-term research objectives. In *Foundations of Augmented Cognition, Third International Conference 2007, LNAI 4565*, ed. D. D. Schmorrow and L. M. Reeves, 220–8. Springler-Verlag Berlin Heidelberg: Germany.

Reeves, L. M., P. Young, D. D. Schmorrow, and K. M. Stanney. 2007b. Near-term, mid-term, and long-term research objectives for augmented cognition: (a) Robust controller technology and (b) mitigation strategies. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, K. M. Stanney, and L. M. Reeves, 2nd ed. Arlington, VA: Strategic Analysis, Inc.

Sajda, P., A. Gerson, and L. Parra. 2003. Spatial signatures of visual object recognition events learned from single-trial analysis of EEG. In *Proceedings of IEEE Engineering in Medicine and Biology Annual Meeting*, 2087–90. Cancun, Mexico. IEEE.

Scerbo, M. W. 2001. Adaptive automation. In *International Encyclopedia of Ergonomics and Human factors*, ed. W. Karwowski, 1077–79. London: Taylor & Francis.

Schmorrow, D. D., and A. Kruse. 2002. Improving human performance through advanced cognitive system technology. In *CD Proceedings of the Interservice/Industry Training, Simulation & Education Annual Conference (I/ITSEC'02)*. Orlando, FL National Training and Simulation Association (NTSA).

Schmorrow, D. D., and A. A. Kruse. 2004. Augmented cognition. In *Berkshire Encyclopedia of Human-Computer Interaction*, ed. W. S. Bainbridge, 54–9. Great Barrington, MA: Berkshire Publishing Group.

Schmorrow, D. D., and A. Kruse. 2005. Session overview: Foundations of augmented cognition. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, 441–5. Mahwah, NJ: Lawrence Erlbaum Associates.

Schmorrow, D. D., and L. M. Reeves. 2007. Introduction: 21st century human-system computing: augmented cognition for improved human performance. *Aviat Space Environ Med (Spec Suppl)* 78(5): B7–B11.

Schmorrow, D., K. M. Stanney, G. Wilson, and P. Young. 2005. Augmented cognition in human-system interaction. In *Handbook of Human Factors and Ergonomics*, ed. G. Salvendy, 3rd ed. New York: John Wiley.

Schnell, T., M. Keller, and T. Macuda. 2008. Sensor integration to characterize operator state. In *Augmented Cognition: A Practitioner's Guide*, ed. D. D. Schmorrow and K. M. Stanney, 41–74. Santa Monica, CA: Human Factors and Ergonomics Society.

Schnell, T., R. Cornwall, M. Walwanis, and J. Grubb. 2009. The quality of training effectiveness assessment (QTEA) tool applied to the naval aviation training context. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. N. D. D. Schmorrow, et al., 640–49. Springer-Verlag Berlin Heidelberg, Germany.

Smith, E. E., and J. Jonides. 1998. Neuroimaging analyses of human working memory. *Proc Natl Acad Sci* 95(20):12061–68.

Smith, M. E., L. K. McEvoy, and A. Gevins. 1999. Neurophysiological indices of strategy development and skill acquisition. *Cogn Brain Res* 7(3):389–404.

Smith, M. E., A. Gevins, H. Brown, A. Karnik, and R. Du. 2001. Monitoring task loading with multivariate EEG measures during complex forms of human-computer interaction. *Hum Factors* 43:366–80.

Stanney, K. M., and L. Reeves. 2005. *Mitigation Strategies and Performance Effects*. White paper outbrief from a working session at Improving Warfighter Information Intake Under Stress, AugCog PI Meeting, March 2–4, 2005. Chantilly, VA.

Stanney, K. M., D. D. Schmorrow, M. Johnston, S. Fuchs, D. Jones, K. S. Hale, A. Ahmad, and P. Young. 2009. Augmented cognition: An overview. In *Reviews of Human Factors and Ergonomics*, ed. F. T. Durso, 5:195–224. Santa Monica, CA: Human Factors and Ergonomics Society.

Sukaviriya, P., and J. D. Foley. 1990. Coupling a UI-framework with automatic generation of context-sensitive animated help. In *Proceedings of the 3rd Annual ACM SIGGRAPH Symposium on User Interface Software and Technology*, 152–66. New York: ACM Press.

Sun, Y., H. Wang, Y. Yang, J. Zhang, and J. W. Smith. 1994. *Probabilistic Judgment by a Coarser Scale: Behavioral and ERP Evidence*. www.cogsci.northwestern.edu/cogsci2004/papers/paper187.pdf (accessed December 5, 2005).

Tremoulet, P., P. Barton, P. Craven, C. Corrado, G. Mayer, and K. Stibler, et al. 2005. *DARPA Improving Warfighter Information Intake Under Stress—Augmented Cognition Phase 3 Concept Validation Experiment (CVE) Analysis Report for the Lockheed-Martin ATL team (Prepared Under Contract No. NBCH030032)*. Arlington, VA: Defense Advanced Research Projects Agency/Information Processing Techniques Office.

Tucker, D. M., and P. Luu. 2009. Operational brain dynamics: data fusion technology for neurophysiological, behavioral, and scenario context information in operational environments. In *Augmented Cognition, HCII 2009, LNAI 5638*, ed. D. D. Schmorrow, et al., 98–104. Springer-Verlag Berlin Heidelberg, Germany.

Vinayagamoorthy, V., A. Brogni, M. Gillies, M. Slater, and A. Steed. 2004. An investigation of presence response across variations in visual realism. In *Presence 2004: 7th Annual International Presence Workshop*, 148–55. http://www.temple.edu/ispr/prev_conferences/proceedings/2004/index.html (accessed June 30, 2009).

Vogel, E. K., and S. J. Luck. 2000. The visual NI component as an index of a discrimination process. *Psychophysiology* 37:190–203.

Whitlow, S. D., and P. M. Ververs. 2005. Scheduling communications with an adaptive system driven by real-time assessment of cognitive state. In *Proceedings of the 11th International Conference on Human-Computer Interaction*. Las Vegas, NV Lawrence Erlbaum Associates, Inc.

Wickens, C. D. 2002. Multiple resources and performance prediction. *Theor Issues Ergon Sci* 3:159–77.

Wilson, C. R. 1992. Applied use of cardiac and respiration measures: Practical considerations and precautions. *Biol Psychol* 34:163–78.

Wilson, G. F., and C. A. Russell. 2006. Psychophysiologically versus task determined adaptive aiding accomplishment. In *Foundations of Augmented Cognition*, ed. D. D. Schmorrow, K. M. Stanney, and L. M. Reeves, 2nd ed., 201–7. Arlington, VA: Strategic Analysis, Inc.

Wilson, G. F., and R. E. Schlegel, eds. 2004. *Operator Functional State Assessment (NATO RTO Publication RTO-TR-HFM-104)*. Neuilly sur Seine, France: NATO Research and Technology Organization.

Wilson, G. F., J. D. Lambert, and C. A. Russell. 2000. Performance enhancement with real-time physiologically controlled adaptive aiding. In *Proceedings of the Human Factors and Ergonomics Society 44th Annual Meeting*, 361–4. Santa Monica, CA: Human Factors and Ergonomics Society.

Yamaguchi, S., S. Yamagata, and S. Kobayashi. 2000. Cerebral asymmetry of the “top-down” allocation of attention to global and local features. *The Journal of Neuroscience 20, RC72 1 of 5*. http://www.ling.uni-potsdam.de/~saddy/web%20papers/Yamaguchi%20assymetry%20and%20attention.pdf. (accessed June 5, 2011).

61 Social Networks and Social Media

Molly A. McClellan, Julie A. Jacko, François Sainfort, and Layne M. Johnson

CONTENTS

61.1 INTRODUCTION

Social media and social networking systems have become ubiquitous phenomena in the daily routines and activities of hundreds of millions of citizens around the world. In many ways these platforms and technologies have supplanted longstanding traditional forms of communication, information sharing, and relationship building. Although a variety of these social tools are heartily embraced by users, they have been extant for only a relatively short period of time and have become much like a thief in the night, stealing traditional modes of commerce, love, war, and survival and upending them to form an exciting and significant influence on the development of humankind. Yet in some ways, the new social milieu that has been catalyzed by the creation of various social media and networks is rudimentary and relatively crude. The basics of functionality and design have been monetized in rare cases and the necessary refinement and further development of these tools stand to complement the progress of societies in many revolutionary ways.

We describe major social media tools based on their popularity and influence as observed from evidence gathered from significant applications that have impacted disciplines, processes, current events, scholarly paradigms, and social contracts. Our presentation is not limited to social networking sites and includes a discussion of progress in professional networks and research networking software that have arisen as innovative collateral and are being leveraged to further the advancements of commerce, research, and scholarship. Opportunities to describe the application of design principles and aspects of human factor engineering are also presented to provide the reader with a foundation on which to build hypotheses for future contributions to allow for improvements in the use, analysis, and application of social media.

61.2 CONTEXT

Computer mediated communication (CMC) systems have been available for several decades. Initially, the primary form of CMC existed as e-mail applications supported by basic text editing software backed by message exchange functionality that provided users linked to a single mainframe computer to communicate asynchronously. As computing evolved from a mainframe-based model to a personal computer-based model, the drive to link users via local area networks and wide area networks gave rise to the realization that CMC systems could be readily connected over networks in a client server fashion. Users began to realize the efficiencies of e-mail; however, ubiquity of CMC systems did not gather significant momentum until the advent of the World Wide Web. To understand the genesis and evolution of social media, one needs to recognize that the proliferation of the personal computer as a commodity and the adoption of its use by the public at large in most of the developed world drove innovative minds to discover means to deploy a variety of tools to the masses.

Social media then emerged and has experienced explosive growth in the past few years. Wikipedia has over 3.5 million pages with descriptions of entities, Flickr has experienced over 5 billion photos, YouTube has 35 hours of videos uploaded to the site each minute, and Twitter users generate 65 million tweets per day (El Abaddi et al. 2011). The ubiquitousness of social media and social online services provides more and more ways for people to connect socially and professionally, while also transforming the ways we think, behave, and share with others. For example, Facebook, MySpace, Twitter, Wikipedia, Blogs, YouTube, tagging in Flickr, and bookmarking in Delicious are all enacting dynamic social change. People with shared interests now have the ability to form groups, defining and reinforcing virtual community structures, which frequently overlap. Early studies (Garton et al. 1997) of computer-supported social networks did not receive as much attention as studies of human–computer interaction, online person-to-person interaction, and computer-supported communication within small groups. This was, in part, due to the state of the technology at the time. More recently, researchers and developers have often focused their studies on the behaviors of single individuals connected within multiple communities. Wang et al. (2010) have taken this a step further and proposed a novel co-clustering framework, which takes advantage of networking information between users and tags in social media to discover overlapping communities to develop a deeper understanding of group evolution.

Some researchers have expressed concern that the ubiquitousness of the global Internet, in general, has weakened local community and stymied local citizenry involvements in particular, by leading people away from meaningful in-person contact. Hampton and Wellman (2003) examined this in the context of a "wired suburb" near Toronto and concluded that the Internet in fact supported neighboring by facilitating discussion and mobilization around neighborhood issues. It was determined that the Internet especially supports increased numbers of contacts with weaker ties. That is, Internet supports the formation of acquaintance-level friendships that are essential to organization and information-sharing at the neighborhood level. In way of follow-up, Wang and Wellman (2010) used two American national surveys to analyze how changes in the number of friends are related to changes in Internet use. The authors found that friendships continue to be abundant among American adults between the ages of 25 and 74 and that they grew from 2002 to 2007. This trend is similar regardless of whether someone is a nonuser of Internet or a light-to-heavy user of Internet. However, heavy Internet users are particularly active, having the most friends both online and offline.

61.3 WEB 1.0, 2.0, AND 3.0

The naming of web 1.0 did not occur until after the concept of web 2.0 was introduced in 2004 to reference the second generation of the World Wide Web. Realistically, you cannot have a 2.0 version without a 1.0 version, so individuals worked backward to define web 1.0. However, creating this definition would prove to be challenging because the changes that led to the term web 2.0 are not technical changes to the Web, but instead it is a reference to applications that facilitate interactive information sharing, interoperability, user-centered design, and collaboration. Additionally, web 1.0 and 2.0 cannot be defined on a timeline since some of the so-called 2.0 techniques were in existence with the launching of the World Wide Web (Strickland 2008a).

Simply stated, web 1.0 sites are considered to be static, not interactive, and proprietary. For example, in web 1.0, a photography studio may create a static profile page about their business that never changes. In web 2.0, this same studio could create a profile on Facebook that can be frequently updated and allow interactions with "friends." In web 1.0, users could locate a dictionary or encyclopedia online and read the information. The interactivity of web 2.0 allows users to update wikis such as Wikipedia with their knowledge. Firefox is an example of non-proprietary web browser for web 2.0. Unlike previous web 1.0 browsers such as Netscape, Firefox source code is available and users can create their own applications or make enhancements. A variety of interactive web 2.0 applications can be found on the Internet. Web-based communities, hosted services, web applications, social networking sites, video sharing sites, wikis, blogs, mashups, and folksonomies are just some examples of the applications.

Web 2.0 has also been utilized for applications such as healthcare. Social media and social networking enable patients, caregivers, families, and healthcare professionals to connect around topics of mutual interest. Examples include communities that form around topics such as (1) common or related health conditions, (2) treatment options, (3) decision making, (4) support groups, and (5) health lifestyle and well-being. Researchers are just now recognizing, and thus further investigating, the value such applications bring to people in the management and experience of their own health or that of a loved one. Social media is transforming traditional patient–clinician relationships, empowering patients to demand more collaborative approaches to care and encouraging patients and clinicians to engage in shared decision making. Social networks have also enabled people to extend their natural circle of social support to include others who may be facing similar circumstances or experiences, but may be at a more comfortable emotional distance (Colineau and Paris 2010).

Cancer survivorship is an area that has realized much attention where social networking is concerned. A diagnosis of cancer is always scary and very often life-changing. YouTube is one tool that has been used by cancer patients to share their personal narrative with others who may be facing a similar diagnosis. Chou et al. (2011) conducted an in-depth characterization of authentic personal cancer stories on YouTube to extract common attributes of those narratives and further illuminate their value to cancer survivors. Their results point to common characteristics of authentic cancer survivorship stories online, such as themes of dramatic tension, emotional engagement, markers of loss of control, a sense of depersonalization, and the unexpectedness of a cancer diagnosis.

There is evidence, too, that online social networking is good for one's well-being. Toma (2010) demonstrated that social networking tools, like Facebook, have self-affirming value. Self-affirmation is the process of bringing awareness to one's own talents, tendencies, goals, treasures, and values. When research subjects spent time on their own Facebook profile pages, as supposed to someone else's, they demonstrate identical behaviors to those who experienced a classic self-affirmation manipulation, namely, displaying fewer ego-protective mechanisms in the face of negative feedback on a task. Other researchers have demonstrated that directed communication between pairs of users on Facebook (e.g., wall posts, comments, and "likes") is associated with greater feelings of bonding, social capital, and lower loneliness (Burke, Marlow, and Lento 2010). Social media has also emerged as an important platform for promoting public health efforts such as public health communication and public health information exchange (Kontos et al. 2010).

There is some concern that social media may be harmful to children and families, primarily due to children's limited capacity for self-regulation and susceptibility to peer pressure and the time social media can take away from traditional face-to-face family interactions. Social media also merits awareness in the contexts of Internet addiction and concurrent sleep deprivation (Christakis and Moreno 2009). The benefits of children and adolescents using social media are also recognized; namely, fostering socialization, communication, and enhanced learning. The minimum age for most social media sites is 13 years. Hence, parents and pediatricians play a critical role in monitoring and informing children about the risks and benefits associated with using social media (O'Keefe and Clarke-Pearson 2011). For more information on web 2.0 in Healthcare, see Sainfort et al. (2011).

Web 3.0, or its synonymous term "the semantic web," is a term coined by Tim Berners-Lee, the man who invented the (first) World Wide Web (Metz 2007). The idea behind the semantic web is that instead of search engines scouring websites for keywords, the browser will understand the information on the web and be able to gather, analyze, and present the data to the user. Instead of a human having to read the webpage, the software will essentially do it for us. To better conceptualize web 3.0, consider the following example. You want to take a date out to a critically acclaimed romantic movie, then an Italian dinner at a restaurant with a terrace, but spend less than $150.00. Using web 2.0 technology, you would have to search for available movies, read reviews, search for Italian restaurants with a terrace, and check pricing information yourself. This could be a timely endeavor because of the separate searches required. With web 3.0, it is anticipated you could enter your search as "critically acclaimed romantic movies near an Italian restaurant with a terrace for less than $150." Your search would be returned with options that meet the above criteria.

Although the technology is not yet mature, many experts agree that web 3.0 will be a more personalized and relevant experience. It is also believed that browsing history will create unique Internet profiles for users and that web 3.0 will use the profile to customize their browsing (Strickland 2008b). This would mean that two different users could search for the same term and wind up with entirely different results.

One of the biggest challenges the semantic web faces is the creation of ontologies. In order for the semantic web to work, programs called software agents will crawl through the web, searching through collections of information called ontologies. These ontologies would have to be detailed and comprehensive, existing in the form of metadata, or information included in web page code unseen by humans, but read by computers (Strickland 2008b). The question remains as to whether developers would want to maintain these complex ontologies.

Despite all the conversation and conferences to define the web, many critics still say that this is a marketing ploy or at the very least inconsequential since there are not actually different versions of the web at this point. Perhaps the numbering of the web is arbitrary, but it has sparked excellent conversations about the future of web-based technology. After all, it is important to know where we have been to determine where we are going.

61.4 DEEP WEB

Most web users use common search engines (Google, Yahoo!, etc.) that index only material found on the Surface Web. At a greater magnitude than the Surface Web is the Deep Web, which has been estimated to be 500 times greater (Wright 2008). These seemingly nonexistent pages are hidden from common search engine view until they are created dynamically as the result of a specific search. The Deep Web is also commonly known as the Invisible Web, Deepnet, DarkNet, Undernet, or the Hidden Web.

Although the Deep Web has existed in parallel to the Surface Web for almost the same length of time, it has not been relevant to most users until the explosion of user-generated content (UGC) and web 2.0 applications. Popular web 2.0 applications such as Wikipedia, Tumblr, Flickr, and YouTube allow both the extensive numbers of contributors and various media content to grow at an exponential rate. This rate of growth is nearly impossible for common search engine index sizes to keep up with.

Social networking sites cause an additional problem for common search engines. Security controls for sites such as Facebook, MySpace, Twitter, and Bloggr allow users to segregate their information between "public" and "friend" or "invite only" views. Because of the various privacy settings, common search engines may not have access to any of this information. This makes web crawling a challenge for users seeking out specific social information, such as medical support groups or blogs.

Based on consumer demand to be able to web crawl UGC, new search engines are being developed that specialize in Deep Web content. Common search engines are attempting to keep up by adding additional document types such as .pdf and .docx to crawl this type of information. One of the most impressive, albeit scary, Deep Web search engines is pipl (http://pipl.com/). This Deep Web search engine scans databases for personal profiles, public records, and other people-related documents that are invisible to regular search engines. Examples of Deep-Web resources pipl is able to search using advanced algorithms are personal profiles, member directories, scientific publications, and court records. An initial attempt at scanning one of this chapter's authors using only first and last name produced private Facebook and MySpace profiles and photos, as well as all previous addresses the author has lived at since birth as well as scientific publications. A subsequent search using the same parameters on another Deep Web crawler, Spokeo (http://www.spokeo.com/), produced the home the author lives in, a photo of it, the current property value, marital status, and number of adults living in the home as well as time living there. If one chooses to subscribe to Spokeo, it is advertised that they could learn additional information including current income, Internet user names, e-mail addresses, and phone numbers.

While the information contained on the Deep Web is now more readily accessed, it is still not the primary source of information for most people. The majority of people searching for information on both old friends and new will often go as deep as entering the persons' name in a social networking site or a Google search. Despite this, both users of the Internet and even nonusers should be aware of the potential for security risks and even identity theft that can arise from information shared on the Internet.

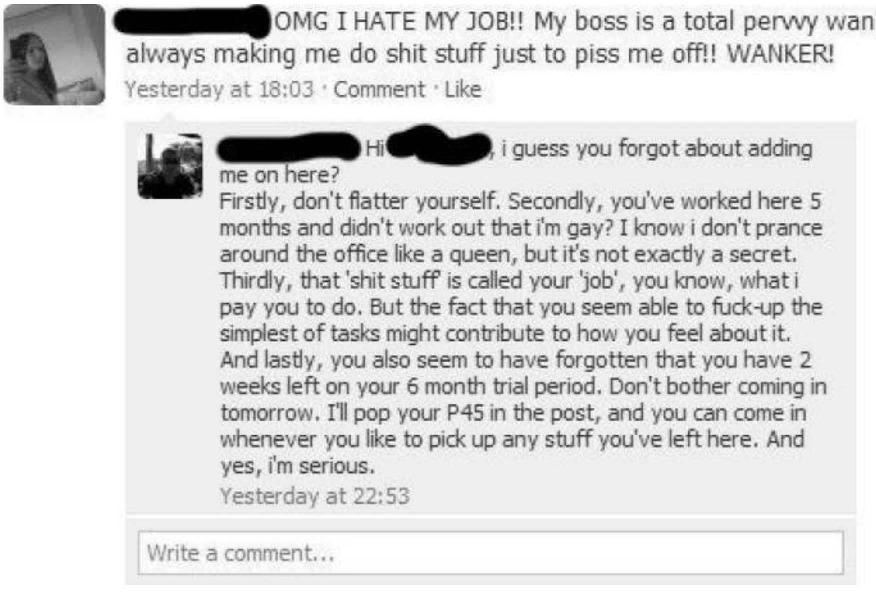
61.5 BE CAREFUL WHAT YOU POST

While social media use is on a rise globally, so is the number of employee layoffs for violating corporate policies. An Internet security firm, Proofpoint, released results of a 2009 survey. They found that of companies with more than 1000 employees, 7% report having issues with employee's use of social media and 8% of those companies report having dismissed someone for posts on Facebook and LinkedIn. In addition, 15% have disciplined an employee for violation of multimedia policies, 13% of U.S. companies investigated an exposure event using mobile or SMS, and 17% disciplined an employee for violating blog or message board policies (Proofpoint 2009). See Figure 61.1 for an example of an employee Facebook post that resulted in job termination. The term "Facebook Fired" has even found its way into the Urban Dictionary (http://www.urbandictionary.com/define.php?term=Facebook+fired) with the definition of "being fired for something you post on Facebook."

Facebook is not the only social media site where employees are finding themselves in trouble for posts. Unfortunately, Twitter's 140 characters or less posting standard makes it even easier than Facebook to blast a statement that results in worldwide shame or loss of a job.

In a now infamous Twitter situation from 2009 known as the "Cisco Fatty" incident, a prospective Cisco employee named Connor Riley innocently tweeted "Cisco just offered me a job! Now I have to weigh the utility of a fatty paycheck against the daily commute to San Jose and hating the work" (quoted in Mangla 2009). Unfortunately for Riley, a Cisco employee named Tim Levad saw her post and tweeted the following in response: "Who is the hiring manager? I'm sure they would love to know that you will hate the work. We here at Cisco are versed in the web" (quoted in Mangla 2009).

While the previous scenario was not an example of an employee losing a job, it caused quite a fuss on Twitter, social media sites, and eventually major media outlets regarding the alleged stupidity and squandering of a job opportunity by not being careful about what one says online. Since then, the number of celebrities, government workers, and everyday people coming under fire or being terminated from employment has risen. Comedian Gilbert Gottfried was terminated in March of 2011 from his contract with Aflac. (Gottfried voiced the Aflac duck for its television commercials.) Gottfried had



[redacted] OMG I HATE MY JOB!! My boss is a total pervy wanker always making me do shit stuff just to piss me off!! WANKER!

Yesterday at 18:03 · Comment · Like

[redacted] Hi [redacted], i guess you forgot about adding me on here?

Firstly, don't flatter yourself. Secondly, you've worked here 5 months and didn't work out that i'm gay? I know i don't prance around the office like a queen, but it's not exactly a secret. Thirdly, that 'shit stuff' is called your 'job', you know, what i pay you to do. But the fact that you seem able to fuck-up the simplest of tasks might contribute to how you feel about it. And lastly, you also seem to have forgotten that you have 2 weeks left on your 6 month trial period. Don't bother coming in tomorrow. I'll pop your P45 in the post, and you can come in whenever you like to pick up any stuff you've left here. And yes, i'm serious.

Yesterday at 22:53

FIGURE 61.1 Employee posting on Facebook, resulting in a termination (From Stewart, T. 2009. Facebook entry that earned 'Lindsay' her P45. *London Evening Standard*. Retrieved November 29, 2011, from http://www.thisislondon.co.uk/standard/article-23732446-facebook-entry-that-earned-lindsay-her-p45.do.)

tweeted over a dozen jokes regarding the Japanese tsunami, and Aflac, Japan's largest insurance company was not amused (Fisher 2011). While many people would have been more sensitive to the tragedy in Japan, many comedians like Gottfried know no boundaries with their jokes. However, even comedians with no ethical code should consider the global reach and impact of their online statements.

Politicians are not immune to potential scandals on social networking sites either. Congressman Anthony Weiner, a democrat from New York, was caught in a Twitter scandal after a photo of a male's genitals was sent from his account via Twitter to a 21-year-old female college student (Kellman 2011). Initially, the congressman denied the allegation that he had sent the photo and claimed his account was hacked. On June 6, 2011, Congressman Weiner held a press conference where he admitted to engaging in inappropriate behavior via social networking sites. He admitted that he had spent the past 3 years sending and receiving dirty online chat messages from six women, which included the exchange of X-rated photos and lewd messages (Fasick and Lisi 2011). This scandal became known almost immediately as Weinergate across all social media sites.

In addition to political scandals, missing job opportunities, or being terminated, social media posters should also be wary of breaking federal laws. Jennifer Carter, a University of Mississippi Medical Center nursing school employee, responded to a tweet from Mississippi Governor Haley Barbour that read "Glad the Legislature recognizes our [state's] dire fiscal situation. Look Forward to hearing their ideas on how to trim expenses" (quoted in Edwards, 2010). Her tweeted response was that Barbour should "Schedule regular medical exams like everyone else instead of paying UMC employees overtime to do it when clinics are usually closed" (quoted in Edwards 2010). Carter was referring to a visit Barbour had made to the medical center several years earlier on a day when it was normally closed. What Carter did not consider was the fact that she potentially violated federal HIPAA regulations through her tweet. While no legal action has been taken against her at this time, she was disciplined and encouraged to resign.

Not only do social media users have to be concerned about what they post publicly, but according to the Maryland Department of Corrections they should be just as worried about their private activities. The American Civil Liberties Union alleged that the state of Maryland required new applicants and those applying for recertification provide their social media account usernames and passwords for use in background checks (Madrigal 2011). To most people it seems reasonable that a current or potential employer would check social media sites to ensure that employees are following a code of conduct or not releasing trade secrets. It seems entirely irrational and unconstitutional that an employer would be able to require access to a private account and seems tantamount to reading your mail (e-mail or paper) or listening in on your phone calls at home.

In reality, despite the public scandals associated with the use of social media, few people wish to share everything, with everyone, all the time. Hence, users often seek to strike a balance between which things to make public and which content to keep private. This also involves how users define "public" within their social sphere, and how to give graduated access to content based on the nature of their relationships. There is a need for design solutions that help users manage the burden of managing privacy and publicness (Lampinen 2010).

61.6 INTERNET DATING AND SOCIAL APPS

The popularity of social networking sites is not limited to maintaining or engaging in new friendships, but has also lead to an increase in online dating sites. Some suggest that because of the economic downturn in the global economy over the past several years, people are fleeing from traditional and more expensive methods of meeting new people (such as meeting at bars, restaurants, or clubs) and instead opting for the less expensive online dating sites. In 2009, it was reported that some of these online dating sites were posting 400% sales growth year on year (Espinoza 2009). Additionally, this billion dollar industry is projected to continue to grow at a rate of 10% annually through 2013 (Tulsiani, Best, and Card 2008). One of the more popular dating sites, Match.com, has seen record growth in 2011. The company's first quarter revenues increased by 18% to $93.3 million, which equates to a 22% increase in subscribers to their website (Silverstein 2011).

In Britain, seven out of ten (69%) dates in 2008 were arranged online through dating sites (Espinoza 2009). In the United States, Internet dating is just as popular. According to a Pew survey, of the ten million Internet users who are single and looking to date, 74% have used the Internet to pursue romantic relationships. The survey also found that approximately 30 million Americans report knowing someone who has been in a long-term relationship or married someone they met online and 60 million people know someone who has engaged in the online dating scene (Madden and Lenhart 2006).

Sautter, Tippett, and Morgan (2010) state that usage patterns of Internet dating services are a result of the following factors: (1) technological change and growing computer literacy making Internet dating available, efficient, and accessible; (2) demographic change causing increased numbers and a variety of persons interested in romantic relationships; and (3) increased acceptability of Internet dating due to social change.

The majority of dating sites such as Match.com or eHarmony operate on similar principles. For a fee, users are allowed to post a profile and photo, search for matches, and communicate through the website. Some of these vendors even offer personality profiling as a feature to suggest personalized matches for the users. Specialized sites have been created for individuals looking for specific types of people, for example, ChristianMingle.com serves to match individuals of Christian faith.

Other sites exist to service people who desire to cheat on their spouses, such as AshleyMadison.com whose slogan is “Life is short. Have an Affair” (http://www.ashleymadison.com/). AshleyMadison even offers an iPhone app called iWipe that will wipe your location data from your iPhone, iPad, or other Apple iOS device so you can prevent others from tracking your whereabouts. Concerns about location tracking were raised when it was discovered that when iPhones were synced to computers, they were building a location database of where users had been. iWipe was released prior to the fix Apple released for consumers to eliminate this location tracking.

AshleyMadison has over 9 million users and assures anonymity to its users. Additionally, if someone leaves their service, AshleyMadison deletes any trace of their account, including messages that they had sent to other members (Kane 2011). Yet another iPhone app that is under fire is from SugarSugar. It is aimed at matching women with rich men, or sugar daddies, and vice versa. Their website sugarsugar.com boasts the motto “Where Romance Meets Finance” and their new app is called “Sugar Lifestyle” (http://www.sugarsugar.com/).

GPS-enabled smartphone apps ignore the complexity of personality profile matching and simplify and speed up dating based on location. Apps like Skout, Grindr, and StreetSpark allow users to search through lists of people based on the users’ GPS location. The apps display the proximity between the user and other people using the app in feet. Users can then exchange messages through the applications to each other and potentially meet face-to-face within minutes. This avoids the need for months of secure online communication and can result in a face-to-face meet within seconds. Although these apps are quite popular, it does raise questions about the safety of meeting a complete stranger based on a few SMSs and due to proximity. Not all users of Skout and Grindr use the GPS feature as both apps allow the ability to turn off the location-aware feature.

61.7 PRIVACY CONCERNS WITH ONLINE DATING

Although online dating sites and apps have seen increases in popularity, there are concerns for safety and privacy. Unlike most social networking sites where friendships are built with colleagues from work, old schoolmates, and friends online, dating brings together two complete strangers with a lack of physical context and verbal cues via the Internet. This can lead to confusion and difficulty in creating and maintaining appropriate relationships. Cyberstalking is another concern for those looking to meet online. As stalkers have increased access to technological tools that allow both intrusion and surveillance into individuals’ lives, society may be making itself more vulnerable to privacy invasion by participating in some of these applications (Spitzberg and Hoobler 2002). There are also concerns over identifying theft and sexual predators lurking online. In April of 2011, a woman filed a civil suit against the popular dating site Match.com alleging that she was sexually assaulted by someone she met online (Associated Press 2011). According to the news article, the suit contends that alleged assailant had faced sex crime charges and the attack could have been prevented if criminal background checks were done by the website. If online dating websites were to begin conducting background screenings, users would then have to pay additional fees and provide the sites with their social security numbers—something many users would not do because of the risk of identity theft.

Another concern with online dating is misrepresentation or skewed self-presentation. In his seminal sociology book, Erving Goffman defines self-presentation as the process of packaging and editing the self to create a certain impression upon the audience (1959). Some instances of misrepresentation are extreme, such as the case of Thomas Montgomery. Montgomery, a 48-year-old married man, posted as an 18-year-old marine to pursue a relationship online with what he believed to be a 17-year-old girl. Montgomery’s wife contacted the girl to tell her the truth. In retaliation, the 17-year-old girl began an online relationship with Montgomery’s 22-year-old coworker. Montgomery then shot and killed his co-worker out of jealousy and pled guilty to the charges in criminal court. The surprising twist was that the 17-year-old girl the coworkers were fighting over was actually a middle-aged housewife engaged in her own online misrepresentation (see Labi 2007).

While extreme cases of misrepresentation exist, research has found that deceptions are usually self-enhancing as opposed to outwardly malicious. For example, male daters have been found to add couple of inches to their height, whereas female daters subtract a few pounds from their weight to appear more attractive to the opposite sex (Toma et al. 2008). According to Hancock and Toma, self-presentational choices are guided by the following: “(a) self-enhancement, or daters’ desire to appear as attractive as possible in order to be noticed by potential mates; and (b) authenticity, or the need to appear honest in their description of themselves” (2010). With improvements in technology, dating profiles are no longer limited to just text. In fact, the profile picture has become a vital component of online dating. Research has found that online daters rated their photos as relatively accurate, but independent judges rated approximately 1/3 of the photographs as not accurate. In addition, female photographs were less accurate than male and were more likely to be older, to be retouched or taken by a professional photographer, and to contain inconsistencies, including changes in hair style and skin quality (Hancock and Toma 2009).

61.8 SOCIAL NETWORKING APPS

The Foursquare app combines social networking with location information by using GPS location, similar to the dating apps in the previous section. Users will “check-in” at a friend’s house, movie theater, or restaurant to display their location. Merchants benefit by adding promotions for people who “check in” to their establishment such as coupons and loyalty programs. Users can bookmark information about venues that they want to visit and find relevant suggestions about nearby venues. Concerned with privacy, Foursquare changed their settings to also allow users to "go off the grid" while still allowing users to receive the benefits of checking-in at a particular merchant (Albanesius 2010). To further engage Foursquare users, in June 2011 a scavenger hunt called "The Clip Trail" powered by AnyClip was launched. Foursquare users play an interactive game to unlock some of the most memorable movie scenes in New York, such as climbing the Empire State Building with King Kong (AnyClip Ltd. 2011). The hunt began during Internet Week New York (IWNY) where users obtain sticker quick response (QR) codes for famous New York film locations. Scanning the QR codes entered them into the scavenger hunt, where they travel to various locations to unlock movie moments for a chance to win 6 months of free movies. This novel scavenger hunt idea takes advantage of Foursquare's open application programming interface (API) and engages social media users in an entirely unique way. We anticipate that more vendors in the future will work to engage consumers through social media apps as the consumer base will demand more engaging and complex experiences.

Meebo is another social media app involved in IWNY 2011. Meebo is an open source platform that integrates all social networks and communications channels into a single, simple-to-use solution (Meebo 2011). It is available on popular social networking sites for instant messaging (although the parent website has likely rebranded their chat but is uses the Meebo API) and mobile devices. Similar to the FourSquare concept, Meebo has a MiniBar that allows users to "check-in" to websites instead of physical locations. As opposed to sharing great new restaurants with your friends, you would share web discoveries with friends and across social networks. It also allows you to follow people with similar interests to see what websites they are visiting. For IWNY 2011, Meebo has launched the "True New Yorker Quest," a guided tour of New York; the twist is that it is for the best New York websites instead of landmarks. By logging in daily during IWNY 2011, users can learn about new websites that give a real feel for the city while having a chance to win prizes (https://www.meebo.com/quest/q/newyork-2011).

61.9 SOCIAL CLOUD

Definitions for cloud computing are changing as rapidly as the technology. Likely the best definition for cloud computing has been defined by the U.S. National Institute of Standards and Technology (NIST). NIST describes cloud computing as a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics (On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured service); three service models (Cloud Software as a Service [SaaS], Cloud Platform as a Service [PaaS], Cloud Infrastructure as a Service [IaaS]); and, four deployment models (Private cloud, Community cloud, Public cloud, Hybrid cloud). Key enabling technologies include (1) fast wide-area networks, (2) powerful, inexpensive server computers, and (3) high-performance virtualization for commodity hardware (2011).

Despite whether web 2.0 users are aware, a large amount of their user data is stored in the cloud. Examples of cloud computing services include Facebook, MySpace, Picasa, Flickr, Bloggr, and YouTube. The advantage of these sites is that users can share their data at any time from any location. This ranges from sharing their blog to utilizing applications on a free social networking platform such as Facebook or Google Docs. The drawbacks are that with the ability to use all these sites, users may lose data and become inconvenienced by their reduced capacity to monitor and manage the data (Seong et al. 2009).

Google appears to be at the forefront of the cloud revolution—with its Android OS and abundant online services it could easily be the hub where everything syncs. Facebook, Amazon, and Microsoft (with its acquisition of Skype) are just a few companies that are also developing new strategies for the cloud. New on the horizon is the Apple announced release of iCloud for Fall 2011. In an attempt to prevent user inconvenience, the iCloud promises to store music, photos, apps, calendars, documents, and more while being able to automatically and wirelessly push them to all your devices (Apple 2011). Apple touts that this will make content management easier, because it will all occur automatically. For example, a picture can be taken with an iPhone and instantly pushed to a Mac or Windows 7 PC. Songs or applications that were previously purchased on the iPhone are now available for the iPad without repurchasing. In addition, users will not have to upload their music to the cloud. Instead, the iCloud will check its vast database, and if it recognizes a song that you own in iTunes, it will automatically add the version to the cloud.

Apple also reports that storage space will not be an issue. Upon signing up for iCloud, users are said to receive 5 GB of free storage. Apple reports that only mail, documents, Camera Roll, account information, settings, and other app data will be applied against that 5 GB of storage. The users-purchased music, apps, books, and Photo Stream will not count against the storage. By taking this route, it encourages the users to use the iCloud for new uses (documents, mail)—not just the typical music and apps most users are familiar with.

Despite Apple's promises about the iCloud, many tech writers and market watchers are skeptical. Apple has released several failed web apps over the past few years, and one author notes that this may just be Apple's Achilles' heel (Fortt 2011). Apple's webmail .me free version lacks essential functionality such as Exchange sync, and the paid version is still not as good as big competitors such as Google who offer superior quality for free. Prior to the iCloud, Apple's experimentation with the cloud was MobileMe, a paid service for $99/year that only provided cloud storage and photo sharing. Similar to the e-mail issue, many free alternatives were available to users and a buggy release of the application only caused it further harm in the public eye. MobileMe is said to be decommissioned with the introduction of iCloud. The final failure for Apple was the launch of Ping, an iTune's social utility that did not integrate well with Facebook, the leading social networking application in the world.

While the iCloud could bode well for Apple if it becomes successful, it may also be yet another signal of the end of the PC as a digital hub. The notion that everything has to be synced back to a PC is antiquated since wireless networks handle this better. While the PC used to be the central device for a tech savvy user, it is clearly now shifting to other devices including smartphones and tablets.

61.10 WIKIS AND TECHNOLOGY-MEDIATED SOCIAL PARTICIPATION

A wiki is defined as "collaborative website comprises the perpetual collective work of many authors. Similar to a blog in structure and logic, a wiki allows anyone to edit, delete, or modify content that has been placed on the website using a browser interface, including the work of previous authors" (Webopedia 2011). It not only refers to the site, but also to the server software that allows users to create and edit content using any web browser.

The most popular and commonly known wiki is Wikipedia, a multilingual web-based free encyclopedia written collaboratively by many users. As of June 9, 2011, the English Wikipedia (http://en.wikipedia.org/) had 3,654,148 articles, 24,147,440 pages in total with 466,735,596 edits. There were 14,708,291 registered users, including 1,790 administrators. Wikipedia is self-governed; any user can add content. However, there are guidelines that should be followed and are available on the site. Because the site is self-organizing, anyone can build a reputation to become an editor. Amongst the editors, there are varying hierarchies including administrators. Despite having over 14 million users, there are less than 2000 admins. These users are allowed to delete articles, block accounts or IP addresses, and edit fully protected articles (http://en.wikipedia.org/wiki/Wikipedia:About).

Although over 14 million registered users are on Wikipedia, even more access their pages to view content but do not edit or add any to the site. A fraction of those users register to become involved in editing and even fewer become editors. Some of those editors will move on to become admins in more of a leadership role. Of those admins, very few excel to roles in governance as Bureaucrats, part of the Arbitration Committee, or Stewards (http://en.wikipedia.org/wiki/Wikipedia:About). Researchers have been trying to build a framework to explain the motivation for this technology-mediated social participation. Preece and Shneiderman proposed the Reader to Leader Framework in 2009 aimed at helping researchers, designers, and managers understand what motivates participation to improve interface design and social support (Preece and Shneiderman 2009).

The Reader to Leader Framework includes some usability and sociability factors that may influence each stage from reader to leader (Figure 61.2). One example of a usability factor that may influence reading is clear navigation paths so that users have a sense of mastery and control, while repeated visibility in online, print, television, and other media is a proposed sociability factor. A usability factor that may lead to contributing would be low threshold interfaces for easily making small contributions, such as no login required. Policies and norms for appropriate contributions are likely sociability factors that would lead to contributing on a given site. Collaboration tools such as the ability to communicate within groups, schedule projects, assign tasks, share work products, and request assistance are usability factors that likely lead to collaboration. Some sociability factors that lead to collaboration may be altruism, respect for one's status within the community, and having an atmosphere of empathy and trust. Usability factors for leaders may include providing enhanced access to promote agendas, expend resources, or limit malicious users. The cultivation and encouragement of leaders is another sociability factor that may influence the creation of that stage (Preece and Shneiderman 2009).

Wikis are often used within and among organizations. Employees within organizations report the use of Wikipedia and other public wikis, private wikis designed for interaction with vendors and customers (Wagner and Majchrzak 2007), and wikis internal to the organization that are used as an information repository for project management and for communication (Grudin and Poole 2010). Researchers collected data from three large companies, online marketing firm, and three start-up software companies. Semi-structured interviews were conducted with employees concerning wiki deployment and use. Three major challenges were identified as barriers to wiki adoption and long-term sustainability: (1) aligning manager and

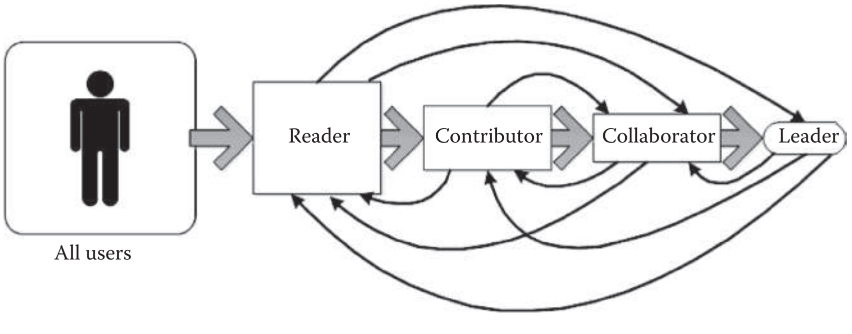


FIGURE 61.2 Reader to Leader Framework. (Adapted from Preece, J., and B. Shneiderman. 2009. The reader-to-leader framework: motivating technology-mediated social participation. *AIS Trans Hum Comput Interact* 1(1):13–32. http://aisel.aisnet.org/thci/vol1/iss1/5.)

individual contributor expectations; (2) content organization and flexibility; and (3) positioning a wiki in an existing information ecology and corporate culture (Grudin and Poole 2010).

61.11 TRENDS IN SOCIAL MEDIA USE

A 2009 Pew Internet Project report reveals that 93% of both teens aged 12–17 and young adults aged 18–29 go online, while 74% of all adults (18+) go online. It is true that while we see increases in Internet use across the generational board (especially increases among ages 65+), the fastest growing Internet users are the millenials and young adults. Social media and web 2.0 applications are extremely popular among younger people, but are also increasingly popular for those over 30. When web 2.0 applications such as Twitter are first launched, we see people of all age categories rushing to be first adopters. As the "newness" of these applications wears off, the demographics for usage are starting to stabilize. These data are now available, and social media usage can explain some of the differences.

A recent report based on the 2009 Parent–Teen Cell Phone Survey, sponsored by the Pew Internet and American Life Project, found a decline in blogging among those under 30, while an increase in blogging activity for those over 30 (Lenhart et al. 2010). One explanation for this decrease in blogging among younger adults is that they may have exchanged blogging for micro-blogging or using applications such as Twitter or Facebook status updates instead of longer, more personal blogs. Another reason may be fear of cyberbullying, as teens realize the more personal information they expose to the world, the greater the possibility of someone using it against them. The increase in blogging for those over 30 may be a result of a slower technology acceptance curve for those who are not digital natives. As content management systems (CMS) such as WordPress continue to provide interfaces that allow templates and widgets to be easily rearranged without knowledge of PHP or HTML, it makes blogging more accessible to everyone. Currently, WordPress has a CMS market share of greater than 50% (W3Techs).

Use of social networking sites has risen dramatically across all age groups. Almost three quarters of both online teens (under 18) and young adults (18–29) use social networking sites. The number of teens has significantly risen from only 55% in November of 2006. Although there is not much difference between teens and young adults, there is a large gap between those groups and adults aged over 30. Only 39% of online adults over 30 use social networking sites (Lenhart et al. 2010).

Although overall usage is up, there is a decline for teens in some of the functionality. Teens are using the social networking sites less for daily messaging, bulletins, and both group and private messages. One can assume that the decline in messaging is due to greater cell phone accessibility for teens—75% of teens now report having a cell phone and 58% of 12 year olds now have phones (Lenhart et al. 2010).

Based on some social networking behaviors, one could assume that concern for privacy is an issue. Many adult social networking site users now maintain two different profiles—over half report having at least two different profiles (Lenhart et al. 2010). Many users maintain two different profiles to keep "work" and "home" life separate. Adults may be worried that having business colleagues view wall postings by friends or pictures they are "tagged" in may lead to embarrassment and possible termination from employment. For this reason, adults may maintain a professional profile as well as a family profile. This may not always be successful as we learn in the section of this chapter on the Deep Web—if the information exists on a social networking site, it may never be truly private.

The most popular social networking site for adults both under 30 and over 30 is Facebook (Lenhart et al. 2010). This statistic is not surprising as Facebook was initially created for college students and later opened up to high school and then everyone over age 13. It is estimated that over 40% of the U.S. population has a Facebook account, one-third of which are aged 24 and younger (Wells 2010). Americans do not just create their social networking accounts, they are actively using them. Nielson Company reports that U.S. Facebook users log in approximately 19 sessions per month and spend an average of around 6 hours per session (The Nielson Company 2010).

Twitter is likely the least popular social media application amongst teenagers. It is interesting to note that although teens aged 12–17 flock to most social media applications, only 8% of them use Twitter for posting status updates or following other users (Lenhart et al. 2010). Teens may not have avidly adopted this situation because they have a wide variety of ways to communicate with each other—text message via cell phone, status updates on their social networking sites, and instant messaging services. They are also likely not politically active and, therefore, do not follow news and politics via Twitter. Likewise, the survey also found that Twitter is similarly less popular for those over 30—only 19% use the application (Lenhart et al. 2010). This age demographic may have social media overload, and although adopting Facebook to keep in touch with old friends may seem reasonable, following what the latest celeb had for breakfast that morning is not that appealing. The lack of popularity may also have to do with the slower rate of technology adoption for older adults. Twitter is the most popular for young adults aged 18–29, of which one-third use Twitter (Lenhart et al. 2010). This popularity may be due to increased interest in being tuned in to the latest news, politics, and celebrity gossip among this age group. It may also be due to increased self interest. This group of people is just coming of age as adults and may be narcissistic enough to believe that other Twitter users would be interested in their thoughts, experiences, and favorite morning breakfast cereals. More research is warranted to determine the true causes behind the age disparities in Twitter usage.

61.12 TEENAGERS AND SOCIAL MEDIA

Teenagers are using social networking sites at an ever increasing rate. Because of the digital divide between teens and adults, many parents are unaware of what their kids are posting or how much time they are spending on these sites. In a national survey of more than 1000 teens, CommonSense

Media discovered that over half of the teens surveyed check their accounts at least once a day, while almost a quarter of them check more than 10 times daily (2009). While avidly checking social networking sites is not a problem for most teens, the behaviors they engage in can be harmful or dangerous. The following results from the teenagers who participated in the survey outline some disturbing trends:

- 37% have made fun of other students.
- 28% have shared personal information about themselves they would not normally share in public.
- 25% have created a profile with a false identity.
- 24% have hacked into someone else's social networking account.
- 13% have posted naked or semi-naked photos or videos of themselves or others online.
- 12% of teens with Facebook or MySpace pages admit their parents do not even know about the account.

The last statistic lends credibility to the assumption that preteens are lying about their ages to sign up for social networking sites. Currently, social networking sites require that users be at least 13 years of age to create an account. This is a minor obstacle for digitally savvy children since there is no method of "carding" or checking IDs for subscription to these sites. Parents may also give permission to their children or assist in creating these false accounts so that their child is not "left out." Allowing underage children to access social networking sites exposes them to situations they may not be prepared for, such as threats of inappropriate content and contact from strangers and bullying by computer (Richtel and Helft 2011).

61.13 CYBERBULLYING

Cyberbullying is an ever increasing form of bullying occurring on school campuses nationwide. Bullying no longer requires face-to-face encounters in the school yard. With the increased access to unsupervised technology, cyberbullys can use e-mail, text, chat rooms, cell phones, cell phone cameras, personally created websites, and social networking sites to taunt and torture their victims.

The importance of this issue was first recognized in 1999 in a report from the United States' Attorney General to the U.S. Vice President Al Gore that suggested incidents of cyber harassment were an increasing problem for law enforcement officials (Beckerman and Nocero 2003). In 2004, a U.S. survey found that 15% of teens identified themselves as Internet bullies, while 7% said they had been targets of cyberbullying (Ybarra and Mitchell 2004). In a Pew Internet Project research survey, approximately one-third (32%) of all teenagers who use the Internet report having experienced receipt of threatening messages, private e-mails or text messages forwarded without consent, an embarrassing picture posted without permission, or having online rumors spread about them online (Lenhart 2007). This survey also found a gender gap for bullying. Girls are more likely than boys to be bullied, and more often it is the older girls (ages 15–17) who experience it the most. Social networking also appears to play a role in frequency of bullying. The survey also found that 39% of teen social network users have been cyberbullied, compared with 22% of teens who use the Internet but not social networks.

More recently, Li (2010) found that 40% of students would do nothing if they were cyberbullied and only about 1 in 10 would inform adults. Reasons for not informing adults include fear of retaliation from the bullies and a feeling of helplessness that anything would be accomplished by reporting the bullying. The increase in incidences of cyberbullying coupled with the lack of reporting makes this a difficult problem for parents, schools, and law enforcement to address.

61.14 SPECIAL POPULATIONS AND SOCIAL MEDIA

Unfortunately, the ability of social networking sites to enhance community ties and structure is not universal across the population. People with learning disabilities or cognitive impairments often encounter substantial barriers when attempting to use them. Older adults also experience challenges when attempting to use such services or avoid adoption altogether.

Some of these difficulties for people with learning disabilities or cognitive impairments are inherent to accessibility and usability problems common to all Internet services. However, this issue also stems from the fact that many simplified versions of popular social networking sites were developed primarily with visually impaired persons in mind or for people using the service over slow network connections or on mobile devices. The problems encountered by people who have learning disabilities or more dramatic cognitive impairments are often related to a lack of comprehension or understanding of the operating environment and the unique terminology/lingo that are used within those environments. In 2009, the Papunet network service unit of The Finnish Association on Intellectual and Developmental Disabilities (FAIDD) launched a project to create an easy-to-use social network service. Principles learned from this effort were applied to the development of a new user interface, based on the Elgg application framework (Elgg 2011) that enabled design and testing of features that render an easily accessible network service. The testing revealed that simplicity and clarity are the keys to design with these populations (Sillanpaa, Alli, and Overmark 2010). A qualitative study involving an existing social networking site and interviews of individuals and groups of older adults revealed that in this population, social networking sites are often perceived to involve socially unacceptable behavior (Lehtinen, Nasanen, and Sarvas 2009).

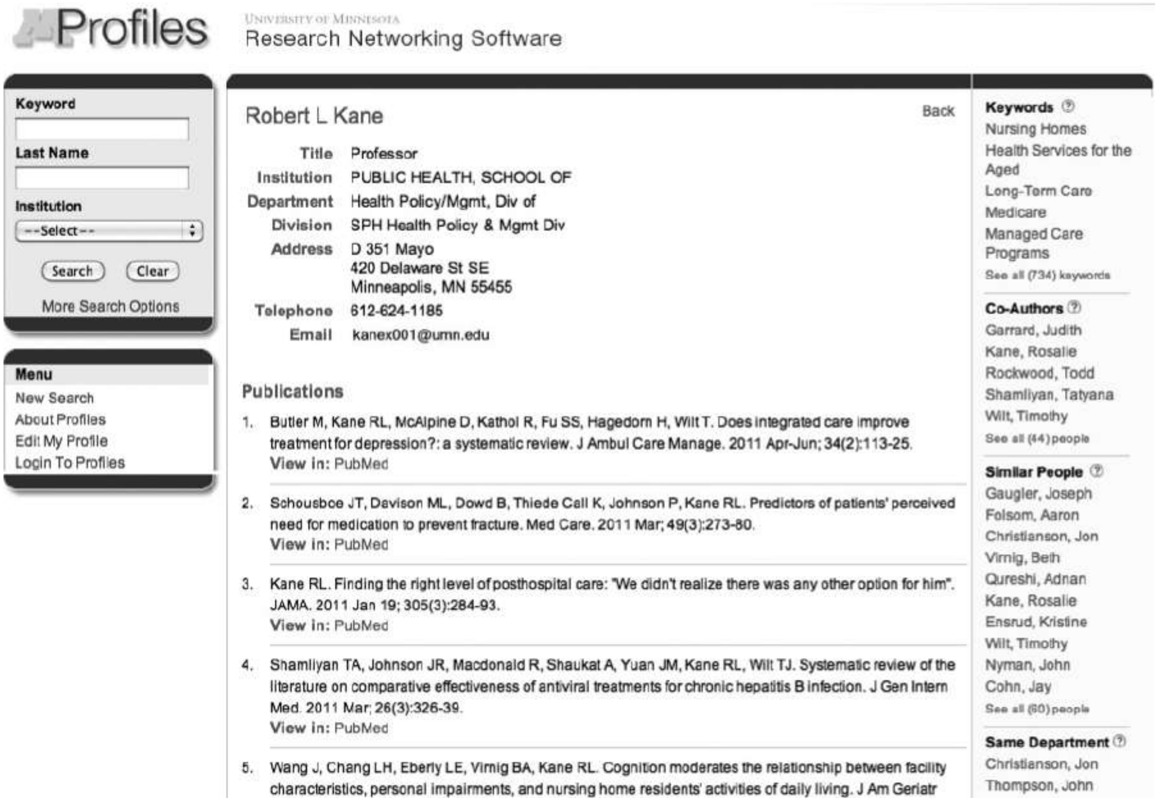
61.15 RESEARCH NETWORKING SOFTWARE

Another form of networking that is enhanced by social media is research networking. Research Networking Software (RNS) is based on the ability to discover resources and people and use data-mining, social networking principles, and semantic web solutions to facilitate the identification of expertise and collaborators (Friedman et al. 2000, Schleyer et al. 2008, Gewin 2010). The identification of collaborators outside of one's primary discipline has become more important as science becomes more interdisciplinary, albeit in small steps (Porter and Rafols 2009). Most recently, convergence has been recognized as the merging of distinct technologies and disciplines into a unified whole, particularly in the engineering, physical sciences, and life sciences (Massachusetts Institute of Technology 2011), and is highly dependent on collaboration among research groups that have traditionally been viewed as distinct and potentially contradictory.

While friends and families share information on Facebook, researchers have various RNS platforms available to connect and transcend disciplines. This approach allows more rapid discovery, leads to the development of collaborative research teams, and provides access to a multitude of resources that are being surfaced through research networking systems. VIVO (vivoweb.org 2011) and Profiles (Harvard Catalyst Profiles 2011) are two open source RNS platforms that allow users to identify experts and collaborators and are being used by a number of research institutions conducting clinical and translational research (Weber et al. 2011). VIVO is an RNS based on a structured information architecture that leverages a number of existing and developing ontologies and relies on semantic web principles including Resource Description Framework (RDF), Web Ontology Language (OWL), SPARQL Protocol and RDF Query Language (SPARQL) to help ensure interoperability with other RNS platforms. Profiles also uses the same structured information architecture to produce open data that can be easily linked to other systems. SciVal is a commercially available RNS (that also provides access to research expertise, collaborators, and other research assets including patents and grant information).

The efficiencies that can be experienced through the use of an RNS include the self-population of a database of publication history, research interests, and professional relationships for investigators in an organization. A representative "snapshot" of a researcher's profile from UMN Profiles (profiles.ahc.umn.edu) is shown in Figure 61.3. Research networking tools most often include visualization capabilities to allow users to view and manipulate dynamic coauthor networks, review publication history over time, analyze geographical interrelatedness of investigators, and probe the degree a researcher collaborates with other investigators. As other forms of visualization develop, it is anticipated that tools such as ResearchWave (Hinrichs, Fisher, and Riche 2010) will be increasingly used to permit a "walk up and use" approach to engage people with information about research at an institution in an aesthetically pleasing manner.

Most recently, a national network of institutions using RNS platforms has been created. The pilot work has been



Robert L Kane

Title Professor
Institution PUBLIC HEALTH, SCHOOL OF
Department Health Policy/Mgmt, Div of
Division SPH Health Policy & Mgmt Div
Address D 351 Mayo
420 Delaware St SE
Minneapolis, MN 55455
Telephone 612-624-1185
Email kanex001@umn.edu

Publications

1. Butler M, Kane RL, McAlpine D, Kathol R, Fu SS, Hagedorn H, Wilt T. Does integrated care improve treatment for depression?: a systematic review. J Ambul Care Manage. 2011 Apr-Jun; 34(2):113-25.
View in: PubMed
2. Schousboe JT, Davison ML, Dowd B, Thiede Call K, Johnson P, Kane RL. Predictors of patients' perceived need for medication to prevent fracture. Med Care. 2011 Mar; 49(3):273-80.
View in: PubMed
3. Kane RL. Finding the right level of posthospital care: "We didn't realize there was any other option for him". JAMA. 2011 Jan 19; 305(3):284-93.
View in: PubMed
4. Shamliyan TA, Johnson JR, Macdonald R, Shaukat A, Yuan JM, Kane RL, Wilt TJ. Systematic review of the literature on comparative effectiveness of antiviral treatments for chronic hepatitis B infection. J Gen Intern Med. 2011 Mar; 26(3):326-39.
View in: PubMed
5. Wang J, Chang LH, Eberly LE, Virnig BA, Kane RL. Cognition moderates the relationship between facility characteristics, personal impairments, and nursing home residents' activities of daily living. J Am Geriatr

Keywords ⓘ
Nursing Homes
Health Services for the Aged
Long-Term Care
Medicare
Managed Care Programs
See all (734) keywords

Co-Authors ⓘ
Garrard, Judith
Kane, Rosalie
Rockwood, Todd
Shamliyan, Tatyana
Wilt, Timothy
See all (44) people

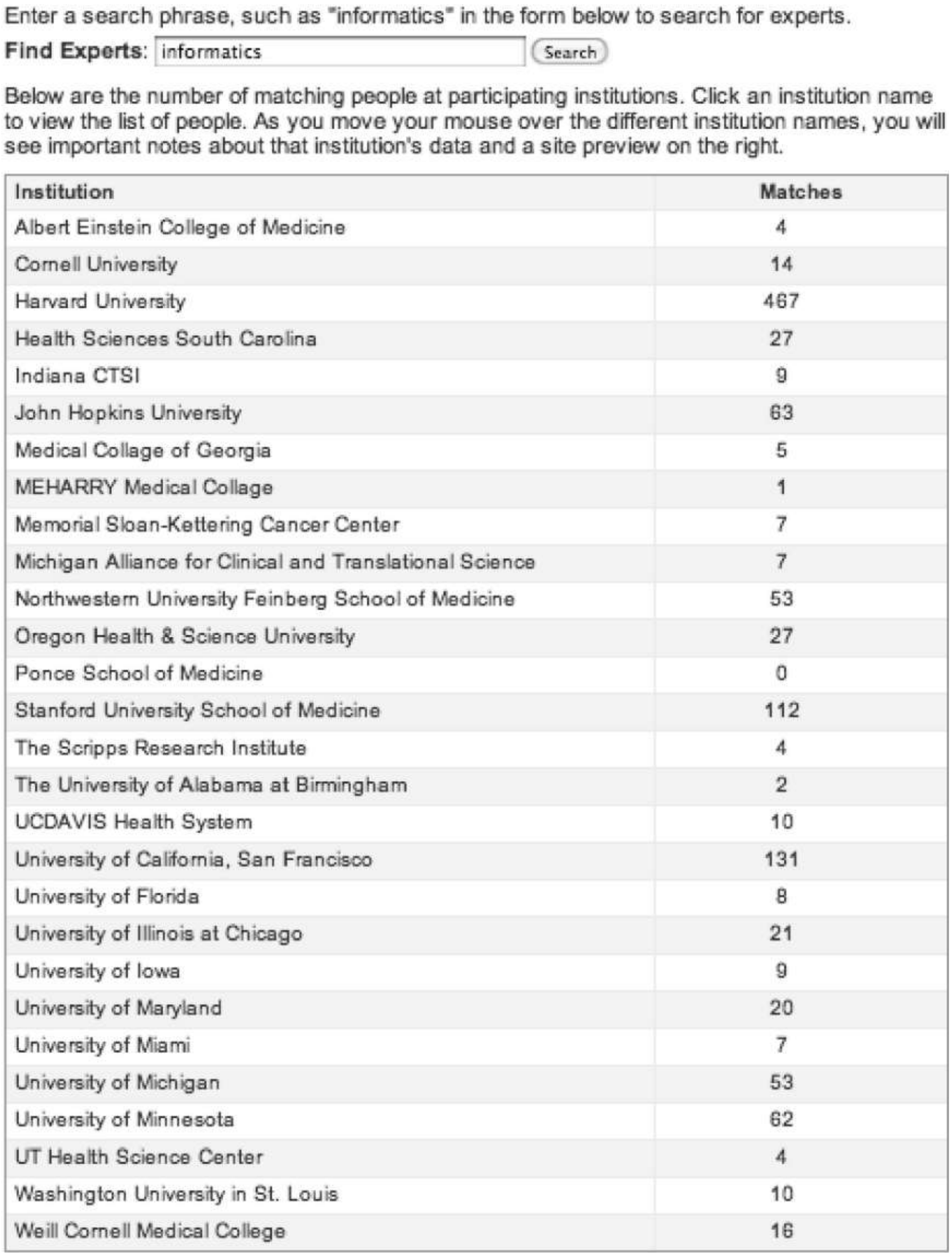
Similar People ⓘ
Gaugler, Joseph
Folsom, Aaron
Christianson, Jon
Virnig, Beth
Qureshi, Adnan
Kane, Rosalie
Ensrud, Kristine
Wilt, Timothy
Nyman, John
Cohn, Jay
See all (60) people

Same Department ⓘ
Christianson, Jon
Thompson, John

FIGURE 61.3 UMN Profiles research networking software generates a researcher's profile that includes directory information and publications that are extracted from PubMed, and on the right side-bar, UMN Profiles research networking software automatically identifies passive networks of related people and concepts that can be further explored to identify potential collaborators.

designated the distributed interoperable research experts collaboration tool (DIRECT), which provides for expertise and collaborator searching across a 28-member network using seven different RNS platforms. Users may search the DIRECT network from inside their own RNS tool or they may also search http://direct2experts.org, which provides a stand-alone interface and other information about DIRECT. A representative results set on the search term "informatics" is shown in Figure 61.4. It is expected that this national network will grow as more institutes adopt RNS platforms as part of their research strategies.

As annotation becomes more commonplace for associating metadata with an object, the opportunity to generate open data for integration will drive the future capabilities of RNS



Enter a search phrase, such as "informatics" in the form below to search for experts.

Find Experts: informatics

Below are the number of matching people at participating institutions. Click an institution name to view the list of people. As you move your mouse over the different institution names, you will see important notes about that institution's data and a site preview on the right.

Institution	Matches
Albert Einstein College of Medicine	4
Cornell University	14
Harvard University	467
Health Sciences South Carolina	27
Indiana CTSI	9
John Hopkins University	63
Medical Collage of Georgia	5
MEHARRY Medical Collage	1
Memorial Sloan-Kettering Cancer Center	7
Michigan Alliance for Clinical and Translational Science	7
Northwestern University Feinberg School of Medicine	53
Oregon Health & Science University	27
Ponce School of Medicine	0
Stanford University School of Medicine	112
The Scripps Research Institute	4
The University of Alabama at Birmingham	2
UCDAVIS Health System	10
University of California, San Francisco	131
University of Florida	8
University of Illinois at Chicago	21
University of Iowa	9
University of Maryland	20
University of Miami	7
University of Michigan	53
University of Minnesota	62
UT Health Science Center	4
Washington University in St. Louis	10
Weill Cornell Medical College	16

FIGURE 61.4 Institutions searchable through distributed interoperable research experts collaboration tool with results for the search term "informatics" listed in the right-hand column.

platforms. The semantic web as the annotation web (Goble et al. 2006) will include RNS platforms as key drivers to promote the development and use of open data and linked open data. The VIVO advances to use linked open data to describe investigators, and their research is being adopted by other RNS platforms to create an RNS-linked open data cloud during 2011–2012.

61.16 SUMMARY

Although we have covered the major social networking tools and discussed progress in professional networks and research networking software, there are many applications and areas of research left untouched. Because of the rapid pace of technology development and the increase in accessibility for broader areas of the population, we anticipate a more dramatic climb in the use of social media applications. Future applications and areas of research would likely include the utilization of social media for national priorities such as healthcare, emergency response, economic development, and education.

Privacy and security issues will continue to be at the forefront as we learn new ways to communicate and share information via the web. New hardware and improvements to smartphones, tablets, and other mobile devices will foster opportunities to engage humans with their social and physical environments in unique ways. The PC will likely become just another device instead of a hub as we continue to look to the cloud for data storage and accessibility. Increased awareness of design principles and aspects of human factor engineering will serve to improve usability of current systems and help define even better applications in the future. Social media and social networking are phenomena that will continue to transform the ways we interact, connect, communicate, share, work, and build communities.

REFERENCES

About Meebo meebo. 2011. *Meebo—Together is Better*. http://www.meebo.com/about/ (accessed June 9, 2011).

Albanesius, C. 2010. Foursquare updates privacy settings—news & opinion—PCMag.com. In *Technology Product Reviews, News, Prices & Downloads—PC Mag.com—PC Magazine*. http://www.pcmag.com/article2/0,2817,2367964,00.asp (accessed June 8, 2011).

Anyclip, LTD. 2011. AnyClip to Kickoff Foursquare Scavenger Hunt During NYC Internet Week. *Marketwire â Newswire service for online press release distribution, social media releases, social media monitoring, online newsrooms, news release analytics and reporting*. http://www.marketwire.com/press-release/anyclip-to-kickoff-foursquare-scavenger-hunt-during-nyc-internet-week-1524476.htm (accessed June 9, 2011).

Apple—iCloud stores your content and pushes it to your devices. 2011. *Apple*. http://www.apple.com/icloud/what-is.html (accessed June 7, 2011).

Associated Press. 2011. California Woman Sues Internet Dating Site After Sex Assault—FoxNews.com. *FoxNews.com—Breaking News—Latest News—Current News*. http://www.foxnews.com/us/2011/04/14/california-woman-sues-internet-dating-site-sex-assault/ (accessed June 9, 2011).

Beckerman, L., and J. Nocero. 2003. High-tech student hate mail. *Educ Dig* 68(6):37–40.

Burke, M., C. Marlow, and T. Lento. 2010. Social network activity and social well-being Social media users. In *Proceedings of the 2010 ACM Conference on Human Factors in Computing Systems*, 1909–12. New York: ACM.

Chou, W. Y., Y. Hunt, A. Folkers, and E. Augustson. 2011. Cancer survivorship in the age of YouTube and social media: a narrative analysis. *J Med Inter Res* 13(1):e7.

Christakis, D. A., and M. A. Moreno. 2009. Trapped in the net: Will Internet addiction become a 21st century epidemic? *Arch Pediatr Adolesc Med* 163:959–60.

Colineau, N., and C. Paris. 2010. Talking about your health to strangers: understanding the use of online social networks by patients. *MYPERMM New Review of Hypermedia and Multimedia* 16 (1/2):141–60.

Edwards, E. 2010. Twitter: Without a doubt, the best way to get fired. *JOLT Home—North Carolina Journal of Law and Technology*. http://www.ncjolt.org/blog/2010/01/14/twitter-without-doubt-best-way-get-fired (accessed April 2, 2011).

Elgg. 2011. http://elgg.org (accessed June 9, 2011).

El Abaddi, A., L. Backstrom, S. Chakrabarti, A. Jaimes, J. Leskovec, and A. Tomkins. 2011. *Proceedings of the 2011 International Conference on the World Wide Web*, 2:327–8. New York: ACM.

Espinoza, J. 2009. Online dating sites flirt with record growth—forbes.com. *Forbes com* http://www.forbes.com/2009/01/06/online-dating-industry-face-markets-cx_je_0105autofacescan01.html (accessed June 8, 2011).

Fasick, K., and C. Lisi. 2011. Weiner says he has no plans to resign following sexting scandal. *New York Post*. Retrieved June 30, 2011, from http://www.nypost.com/p/news/local/weiner_says_scandal_has_no_plans_RE26B787X2Xp2K4Le8qIeK.

Fisher, M. 2011. Gottfried's QuackerQuashed, fired for tweets. *The Faster Times*. thefastertimes.com/entertainmentnews/2011/03/16/gottfrieds-quacker-quashed-fired-for-tweets/ (accessed April 1, 2011).

Fortt, J. 2011. CNBC Tech Check Blog—Fortt: Why Apple's iCloud Announcement Matters—CNBC.com Technology News—CNBC. *Stock Market News, Business News, Financial, Earnings, World Market News and Information—CNBC*. http://www.cnbc.com/id/43294274 (accessed June 7, 2011).

Friedman, P. W., B. L. Winnick, C. P. Friedman, and P. C. Mickelson. 2002. Development of a MeSH-based index of faculty research interests. *Proc Am Med Inform Assoc Symp* 265–9. Bethesda, MD: AMIA.

Garton, L., C. Haythornewaite et al. 1997. Studying online social networks. *J Comput Mediat Commun* 3(1):0–10.

Gewin V. 2010. Collaboration: Social networking seeks critical mass. *Nature* 15(468):993–4.

Goble, C., O. Corcho, P. Alper, and D. De Roure. 2006. e-Sciene and the semantic web: A symbiotic relationship. *Proc Discov Sci Lect Notes Artif Intell* 1–12.

Grudin, J., and E. S. Poole. 2010. Wikis at work: Success factors and challenges for sustainability of enterprise wikis. In *ACM Press: Proceedings of WikiSym* '10. Gdansk, Poland. New York: ACM.

Hampton, K., and B. Wellman. 2003. Neighboring in netville: how the Internet supports community and social capital in a wired suburb. *City Community* 2(4):277–311.

Hancock, J., and C. Toma. 2009. Putting your best face forward: The accuracy of online dating photographs. *J Commun* 59:367–86. http://faculty.unlv.edu/drums/pioneer/files/JOUR%20435%20635/photosonlinedating.pdf (accessed June 9, 2011).

Harvard Catalyst Profiles. 2011. *People and Collaboration*. http://connects.catalyst.harvard.edu/PROFILES/search (accessed on April 4, 2011).

Hinrichs, U., D. Fisher, and N. H. Riche. 2010. ResearchWave: an ambient visualization for providing awareness of research activities. In *DIS '10: Proceeding of 8th ACM Conference on Designing Interactive Systems,* 31–4. New York: ACM.

Kane, Y. I. 2011. New App Aims to Erase Swingers' Online Tracks—Digits—WSJ. *WSJ Blogs—WSJ*. http://blogs.wsj.com/digits/2011/05/03/new-app-aims-to-erase-swingers-online-tracks/ (accessed June 8, 2011).

Kellman, L. 2011. Top Democratic women dodge tough call on Weiner—Politics—Wire—TheSunNews.com. *TheSunNews.com—MyrtleBeachNews,Golf,Hotels,Homes,Jobs,Cars*.N.p., 9 Web. http://www.thesunnews.com/2011/06/09/2211049/weiner-abides-despite-new-photo.html (accessed 9 June 2011).

Kontos, E. Z., K. M. Emmons, E. Puleo, and K. Viswanath. 2010. Communication inequalities and public health implications of adult social networking site use in the United States. *J Health Commun* 15 Suppl (3):216–35.

Labi, N. 2007. An IM Infatuation Turned to Romance. Then the Truth Came Out. *Wired*. http://www.wired.com/images/press/pdf/flirting.pdf (accessed June 9, 2011).

Lampinen, A. 2010. Practices of balancing privacy and publicness in social network services. Poster session II—doctoral colloquium. *GROUP '10: Int Conf Support Group Work* 11-06:343–4.

Lehtinen, V., J. Näsänen, and R. Sarvas. 2009. A Little Silly and Empty-Headed—Older Adults' Understandings of Social Networking Sites in the *23rd BCS conference on Human Computer Interaction*. Cambridge, UK.

Lenhart, A. 2007. Cyberbullying and Online Teens. *Pew Research Center's Internet & American Life Project*. http://www.pewinternet.org/Reports/2007/Cyberbullying.aspx (accessed April 3, 2011).

Li, Q. 2010. Cyberbullying in high schools: A study of students' behaviors and beliefs about this new phenomenon. *J Aggress Maltreat Trauma* 19(4):372–392.

Madden, M., and A. Lenhart. 2006. Most online Americans who are single and looking for dates have used the Internet to pursue their romantic interests and millions more Americans know people who have tried and succeeded at online dating. *Pew Inter Am Life Proj*. http://www.pewinternet.org/Reports/2006/Online-Dating/01-Summary-of-Findings.aspx (accessed June 8, 2011).

Madden, M., and A. Lenhart. 2006. Summary of Findings | Pew Internet & American Life Project. Pew Research Center's Internet & American Life Project. Retrieved November 29, 2011, from http://www.pewinternet.org/Reports/2006/Online-Dating/01-Summary-of-Findings.aspx.

Madrigal, A. 2011. Should employers be allowed to ask for your Facebook login? In *The Atlantic.* http://www.theatlantic.com/technology/archive/2011/02/should-employers-be-allowed-to-ask-for-your-facebook-login/71480/ (accessed April 3, 2011).

Mangla, I. 2009. Fired for Facebook: Don't let it happen to you. *CNN Money*. http://moremoney.blogs.money.cnn.com/2009/04/21/fired-for-facebook-dont-let-it-happen-to-you/ (accessed April 2, 2011).

Massachusetts Institute of Technology. 2011. *The Third Revolution: The Convergence of the Life Sciences, Physical Sciences, and Engineering*. http://web.mit.edu/dc/Policy/MIT%20White%20Paper%20on%20Convergence.pdf (accessed March 24, 2011).

Metz, C. 2007. Web 3.0. PCMag.com. In *Technology Product Reviews, News, Prices & Downloads. PCMag.com. PC Magazine*. http://www.pcmag.com/article2/0,2817,2102852,00.asp (accessed June 9, 2011).

Nielsen Company. 2010. Global Audience Spends Two Hours More a Month on Social Networks than Last Year. In *Nielsen Wire*. blog.nielsen.com/nielsenwire/global/global-audience-spends-two-hours-more-a-month-on-social-networks-than-last-year/print/ (accessed April 2, 2011).

O'Keeffe, G. S., K. Clarke-Pearson, and C. O. Communications. 2011. Clinical Report. The impact of social media on children, adolescents, and families. *Pediatrics*, Retrieved from http://pediatrics.aappublications.org/content/early/2011/03/28/peds.2011-0054.abstract.

Porter, A. L., and I. Rafols. 2009. Is science becoming more interdisciplinary? Measuring and mapping six research fields over time. *Scientometrics* 80(3):710–45.

Preece, J., and B. Shneiderman. 2009. The reader-to-leader framework: motivating technology-mediated social participation. *AIS Trans Hum Comput Interact* 1(1):13–32. http://aisel.aisnet.org/thci/vol1/iss1/5

Richtel, M., and M. Helft. 2011. Facebook users who are under age raise concerns. In *The New York Times*. B1. http://www.nytimes.com/2011/03/12/technology/internet/12underage.html (accessed April 2, 2011).

Sainfort, F., J. A. Jacko, M. McClellan, K. P. Moloney, and V. K. Leonard. 2011. E-health in health care. 2nd ed. In *The Handbook of Human Factors in Web Design*, ed. K.-P. L. Vu and R. W. Proctor, 608–609. Boca Raton, FL: CRC Press.

Sautter, J. M., R. M. Tippett, and S. P. Morgan. 2010. The social demography of Internet dating in the united states. *Soc Sci Q* 91:554–75.

Schleyer T., H. Spallek, B. S. Butler, S. Subramanian, D. Weiss, M. L. Poythress, P. Rattanathikun, and G. Mueller. 2008. Facebook for scientists: requirements and services for optimizing how scientific collaborations are established. *J Med Internet Res* 10(3):e24.

Seong, S. W., S. Hangal, C. Brigham, D. Sengupta, G. Bayer, and J. Seo, *et al.* 2009. *A Distributed Social Networking Infrastructure with Personal Cloud Butlers*. http://gbayer.com/stanford/prpl/www2009_submission_872.pdf (accessed June 7, 2011).

Sillanpaa, N., S. Alli, and T. Overmark. 2010. Easy-to-use network service. ICCHP 2010. *LNCS* 6179:544–49.

Silverstein, E. 2011. Match.com proposes closer partnership with meetic dating site. In *Smarter News, Analysis & Research Communities*. http://www.tmcnet.com/topics/articles/180811-matchcom-proposes-closer-partnership-with-meetic-dating-site.htm (accessed June 8, 2011).

Spitzberg, B. H., and G. Hoobler. 2002. Cyberstalking and the technologies of interpersonal terrorism. *New Media Society* 4:71–92.

Stewart, T. 2009. Facebook entry that earned 'Lindsay' her P45. London Evening Standard. Retrieved November 29, 2011, from http://www.thisislondon.co.uk/standard/article-23732446-facebook-entry-that-earned-lindsay-her-p45.do).

Strickland, J. 2008a. Is there a Web 1.0? HowStuffWorks.com. http://computer.howstuffworks.com/web-10.htm (accessed June 9, 2011).

Strickland, J. 2008b. How Web 3.0 Will Work. HowStuffWorks.com. http://computer.howstuffworks.com/web-30.htm (accessed June 9, 2011).

Toma, C. 2010. Affirming the self through online profiles: beneficial effects of social networking sites using your social network. *Proc ACM CHI 2010 Conf Hum Factors Comput Syst*. 1:1749–1752.

Toma, C., J. T. Hancock, and N. Ellison. 2008. Separating fact from fiction: An examination of deceptive self-presentation in online dating profiles. *Pers Soc Psychol Bull* 34:1023–36.

Tulsiani, B., M. Best, and D. Card. 2008. US Paid Content Forecast, 2008 to 2013. Jupiter Research. http://www.forrester.com/rb/Research/us_paid_content_forecast,_2008_to_2013/q/id/53215/t/2. (accessed June 8, 2011).

VIVO. 2011. VIVO: enabling national networking of scientists. http://vivoweb.org/ (accessed April 4, 2011).

W3Techs. 2011. Usage statistics and market share of content management systems for websites. *W3Techs—World Wide Web Technology Surveys*. http://w3techs.com/technologies/overview/content_management/all (accessed April 3, 2011).

Wagner, C., and A. Majchrzak. 2007. Enabling customer-centricity using wikis and the wiki way. *J Manag Inf Syst* 23(3):17–43.

Wang, H., and B. Wellman. 2010. Social connectivity in America: changes in adult friendship network size from 2002 to 2007. *Am Behav Sci* 53(8):1148–69.

Wang, X., L. Tang, H. Gao, and H. Liu. 2010. Discovering overlapping groups in social media. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM 2010)*. Sydney, Australia. Los Alamitos, CA: CPS.

Weber, G. M., W. Barnett, M. Conlon, D. Eichmann, W. Kibbe, H. Falk-Krzesinski, M. Halaas, et al. 2011. Distributed Interoperable Research Experts Collaboration Tool (DIRECT). *J Am Med Inform Assoc*.

Wells, R. 2010. 41.6% of the US Population has a Facebook Account. *Social media news, strategy, tools, and techniques: Social Media Today*. http://socialmediatoday.com/index.php?q=roywells1/158020/416-us-population-has-facebook-account (accessed April 3, 2011).

Wright, A. 2008. Searching the Deep Web. *CACM* 51(10):14–15.

Ybarra, M. L., and K. J. Mitchell 2004. Youth engaging in online harassment: Associations with caregiver-child relationships, Internet use, and personal characteristics. *J Adolesc* 27:319–36.

62 Human–Computer Interaction for Development

Changing Human–Computer Interaction to Change the World

Susan M. Dray, Ann Light, Andrew M. Dearden, Vanessa Evers, Melissa Densmore, Divya Ramachandran, Matthew Kam, Gary Marsden, Nithya Sambasivan, Thomas Smyth, Darelle van Greunen, and Niall Winters

CONTENTS

62.1 INTRODUCTION

This handbook is about the design of human–computer interaction (HCI) and of solutions for specific applications and domains. This chapter* is about how you can go beyond creating interfaces or improving user experience and make a difference in the world. That difference might be global, such as by doing work that supports the UN's Millennium Goals, or it might be local in ways that are meaningful only to a particular community. The funding for the work may come from a government, a non-governmental organization (NGO), a research council, a charity, or a company; the budget may vary from the price of a ticket to millions of dollars. The people or group for which you hope to make a difference might be on the other side of town or on the other side of the world. Regardless, this chapter is about applying what we know about user-centered design (UCD) to worldwide economic and community Development.

62.1.1 Introducing HCI4D

HCI for Development, or HCI4D, is a new multidisciplinary field. It is still defining itself, as is witnessed by the number of names under which it goes: HCI4D, UCD for Development, Human-centered design for Development, Interaction Design and International Development, and others. While there are nuances that distinguish these, we are using the term "HCI4D" in this chapter to encompass them all. This field is being developed by academics and practitioners, by people all over the world in "developed" and "developing" countries, and by young researchers and senior researchers and practitioners, often working hand-in-hand together in the creation of this exciting new area. Furthermore, consider the following:

- HCI4D is interdisciplinary. Therefore, there is no "one way" or one interpretation that is the only "correct" one. There is discussion and sometimes controversy, but no specific orthodoxy.
- We come from both applied and academic settings. We recognize that both have something important to offer and each has its own set of limitations. We are committed to dialogue across what is sometimes a barrier.
- HCI4D is young and is still changing and defining itself.
- HCI4D is deeply rooted in fieldwork. This brings with it a certain set of responsibilities that all fieldwork does, but they are heightened by the emphasis on Development.
- HCI4D is international. We have colleagues on all continents and have worked on as many as six continents.
- HCI4D focuses on the larger goal of making the world a better place.

The authors of this paper reflect some of this diversity: We include academics and practitioners from four continents, who count themselves as researchers, teachers, consultants, and practitioners. We have all worked in countries other than our own and have dealt with challenges big and small to do so. We represent different worlds on many dimensions. However, there are other voices that are not explicitly part of this chapter, although they are no less important: people in NGOs, people in government Development offices, people in foundations, and perhaps most important, people in local communities with whom we work. We have tried to represent these perspectives as faithfully as we can, but realize that this would have been a different chapter had we been able to include their voices directly.

62.1.2 Terminology

Because HCI4D is interdisciplinary, it is important to define some terms that have multiple different meanings. Additionally, several of these terms are also politically charged, which makes it particularly important that we are clear about what meanings we are ascribing in this chapter.

62.1.2.1 Development

The term "development" has a number of different meanings. In the technology world, "development" refers to the actual creation of a program or design. There are several related

* This chapter has been a group project. Some of the case studies have been provided from additional HCI practitioners in the field. The material was edited by Susan Dray, Ann Light, and Vanessa Evers.

uses of the term in information technology (IT) organizations. Software development, application development, and platform development are among the terms that IT people use to describe different aspects of this process of creation. A second meaning of the term relates to the larger technological process, usually described as Research & Development, which includes earlier phases of planning and investigation ("Research") to guide the actual development. However, neither of these is the meaning that we are stressing in HCI4D. In this chapter, Development refers to broad interdisciplinary efforts and transformative projects to address specific human problems and to influence the "complex economic, social, and political processes of change in countries in the poorer parts of the world" (Oxford University 2010). We will distinguish this meaning of the word from the previous two by capitalizing the initial letter. We will discuss the meaning of this term in more detail below both because it is a new concept for the HCI community and because it is far more contentious and politically charged than the more technically focused meanings of the word.

62.1.2.2 Developing World

In addition to clarifying the term "Development," there is also the question of how to refer to different regions of the world. There are simply no good terms for describing the so-called developed and developing regions. Many older readers will remember the first world/third world distinction that has almost disappeared. Nowadays, we have a range of ways of talking about parts of the world, which reflect economic differences in different regions. "Developing countries" is a prevalent term, but the more accurate description might be "emerging economies" or "developing regions." For the remainder of this chapter, we use the terms "developing/developed" regions to distinguish parts of the world that are distinct from each other in terms of access to resources and so on, but in settling on these two terms (obviously related to Development) to discuss HCI4D, we are merely choosing the best tools that we have for the job of making this distinction. Like so many people working in this area, we have yet to find a way of talking about it that captures the nature of the exchange that we would like to foster.

Last, but certainly not least, it is important to be clear when we describe the main actors in HCI4D.

62.1.2.3 Practitioners

For the purposes of this chapter, "practitioners" are people working within industry. They may do applied research, innovation, design, and/or development. We will refer to their work as HCI4D projects, even though they may do research.

62.1.2.4 Researchers

For the purposes of this chapter, "researchers" are people who work and/or teach in academic settings (or quasi-academic settings such as Corporate Research Labs within companies). We will refer to their work as HCI4D research, even though they may also do design and/or development as part of their research.

As mentioned earlier, we are not explicitly differentiating people in NGOs, government, or other settings. They may work to develop policy or may oversee projects, whether done by researchers or by practitioners.

62.1.3 Related Disciplines

Like HCI, HCI4D has interdisciplinary roots. As Ho et al. (2009) pointed out, these make it challenging to get a complete picture of its history and literature, as well as to define the field from the different mindsets of some of the contributing subdisciplines. This is both a strength and a weakness. Because there are different perspectives, there is a potential for rich dialogue and for collaborative discussion of how to solve problems in holistic ways. This is potentially very powerful. But it is also complex, messy, and very challenging. It makes it very difficult to reach consensus on what is "in" and what is "out" of an area like HCI4D. The risk is always that the holistic and collaborative efforts to solve problems will not be realized in endless discussions of the definition of the area, disagreements about which methods are "best," and which world view is "right."

Most current HCI4D professionals come from a HCI background and are, therefore, aware of these interdisciplinary challenges since they are also present in HCI. However, the addition of Development adds a complexity beyond the challenges within HCI. Within the academic discipline of Development studies, there is a great deal of debate as to what exactly "Development" is and what is the "proper" frame of reference (Economic growth? Post-colonial? Marxist? To name a very few). These debates are deeply political and can be quite contentious. It is beyond the scope of this chapter to describe these in detail, but we would refer the interested reader to Ho et al (2009) for references to the relevant literature.

HCI4D is closely intertwined with Information and Communication Technology (ICT) for developing regions. While the debate on the utility of ICTs in developing countries has largely been won, the challenge of how best to use ICTs for poverty reduction remains (Avgerou and Walsham 2000). Although the link between economic growth and ICT has been established (Kraemer and Dedrick 2001; Jalava and Pohjola 2002), the exact process of how ICTs can be used for poverty reduction in developing countries needs exploration and is open to challenge. In particular, there are few if any theoretical explanations as to how ICTs can assist in building human capacity for poverty reduction in a developing country setting. A strong correlation exists between the access to education and knowledge and poverty indicators such as infant mortality, family size, and women's health (Marker, McNamara, and Wallace 2002). Other studies have also established a close link between poverty and an information gap of the poor (see, e.g., Humphrey 2006). ICT Development projects suffer from high project failure rates (United Nations Asian and Pacific Training Centre for Information and Communication Technology for Development 2010). Therefore, it is clear to us that there is a need for greater emphasis on participative processes of design, greater engagement with potential beneficiaries, and more in-depth and culturally relevant ways of making ICTs available, useful, and usable to a diverse population of people.

62.1.4 From Improving the Interface to Improving the World

Most of the HCI methods and processes described in this handbook are designed to be of service in the development of technology. The term HCI defines the interaction between people and machines and what they can do together, but not why they are interacting. We may have a goal beyond making better interfaces and better means of interaction or improving the interface may be that there is an end in itself, but the term HCI implies nothing about the technology that is being improved or for what reason.

A key difference in the work discussed here is that researchers and practitioners doing HCI4D are not just applying techniques to improve digital tools. They are looking beyond the effectiveness of the tool to the impact that it has on a society, most often in terms of the specific individuals, groups, and communities being studied. What distinguishes HCI4D work and makes it so interesting are these additional goals implying a purpose. If it is "for/4" Development, then it has social and moral intentions to change life in particular ways. Development is inherently and inevitably a political and moral activity—in other words, it involves ethics, values, and beliefs—as well as an economic one. The challenge for HCI4D as a field then is how to go beyond the International Standards Organization (ISO) notions of what constitutes a "successful" technology and incorporate some of the "success" criteria from work in the Development community. This involves more than simply adding a few new criteria from the Development literature to the checklists for evaluating our studies. Criteria for success will be different for different communities, and there may be cultural differences between communities that make it impossible to develop global standards. Finding new ways to describe what is a "successful project" increases the complexity of the questions we are asking and the time needed to understand what we need to achieve. It alters how we relate to the people who will use the technology and the methods we use to engage them in the design process. HCI4D projects can take differing forms, from studies of the appropriation of technology, which can inform future application development, to projects that are more interventionist in nature. But if we come in from outside a community to do HCI "for/4" Development, we are coming in to do more than explore what would constitute well-designed technology for our hosts. We are also assessing aspects of their lives and how this might be improved.

As Kleine and Unwin (2009) discussed in looking at the related field of ICT and Development (ICTD): "Rather than the 'and' of ICTD, the 'for' of ICT4D forces users of the term to confront the moral and political agendas associated with 'Development'." While HCI4D researchers are very knowledgeable about their research agenda, they may not have as good an understanding of Development studies and the associated subtleties of Development practice. If we take the "4" seriously, confronting "moral and political agendas" requires understanding of these issues. Ho et al. (2009) point to the serious responsibilities on HCI4D researchers:

> Thus the acronym "HCI4D," as our community has adopted it, carries a level of intent and purpose. As a community, we do not seek merely to understand how humans and computers interact in developing regions, but are concerned with applying this understanding to improve lives, livelihoods, and freedoms for people in these regions. (ibid)

In the majority of HCI4D projects, we work with local communities as equals, generating local capacity so that participation works, defining Development as a joint endeavor, or supporting existing local initiatives. In doing so, we indicate our interest and our responsibility in understanding the social and political agendas at work and the wider context in which our research is taking place. Therefore, our remit for research is broader and deeper than relating Development to economic growth alone. Development is associated with empowerment through the transforming power of technology and its related activities.

62.1.4.1 "D" in HCI4D

So, the "D" in HCI4D stands for Development, but even this is not straightforward. The tradition that Development grows out of is rooted in the idea that some parts of the world are more developed than others and that these areas have a responsibility to address the gap. It is not altogether distinct from the system of beliefs that allowed parts of Europe to run all over Africa, Asia, and the Americas, ostensibly bringing "civilization" to the world, and the United States ostensibly bringing "peace" by doing so more recently in Vietnam, Afghanistan, and Iraq. Meanwhile, the blatant self-interest of much of that history has left its legacy in the economic conditions in many "developing" countries, where production and trade arrangements benefitted those imposing the terms. This colonial past hangs on in the terminology "developing" and also in some of the attitudes to be found in the field, often much to the indignation of people receiving aid. Many practitioners from "developing" countries have argued that Development too often involves remote and powerful decision makers imposing their idea of what is best on other people and places, often on the basis of very limited information. The World Health Organization talks about *Development cooperation* to throw attention on the collaborative nature of all aid activities and the need for a partnership between donor and recipient. Proponents of participatory Development strategies such as Participatory Action Research (Fals-Borda 1987) and Participatory Rural (or urban) Appraisal (Chambers 1994) aim to enable marginalized people to give voice to their concerns, to participate in decisions that affect them, and to exercise their rights.

An underreported but important side-effect of Development cooperation is that researchers and practitioners in the field learn from the people they are working with. Most often, it is not a case of pouring knowledge into communities, even when many resources come from outside. People have been solving problems successfully using the resources available to them for thousands of years. Local knowledge, skills, and history are imperative to understand patterns that will influence a Development project as well as provide sustainable solutions to problems. In developing these solutions, the external researchers and practitioners gain knowledge and experience that they would not have otherwise, and this knowledge can be used to solve problems in other settings also.

62.1.4.2 Little More on the History of Development to Put HCI4D in Context

The idea of Development as a policy goal can be traced to the inaugural speech of American President Truman in 1949. The vision he declared was of, "…making the benefits of our scientific advances and industrial progress available for the improvement and growth of underdeveloped areas." This has been interpreted as the United States' response to the segregation that accompanied the Cold War and a move to limit the threat of Communism (Truman 1949).

At that time, Development was widely interpreted in terms of increasing nations' economic output. However, this view is now widely recognized as inadequate for a range of reasons. First, it assumes that increasing average incomes will benefit all the population, whereas huge inequalities may not be addressed. Second, it assumes that increasing economic production and consumption is a fundamental good in itself, whereas it often has damaging environmental impacts and might be achieved in the context of extremely repressive regimes. Third, it frames Development of a process of making "underdeveloped" areas more like the "developed world," implying that the way of life in these "developed" countries represents the "best of all possible worlds."

In contrast, the concept of "sustainable Development," upon which much HCI4D work is more or less implicitly based, draws attention to the need to consider the whole technical, financial, social, and environmental "ecosystem" in which a Development initiative is sited. Sustainable Development is in part a response to the failure of projects where technical equipment was given by government donors (or often purchased on the basis of a long-term loan). When the equipment requires maintenance, or parts need replacing, the recipients do not have the skills, the infrastructure, or the currency to solve the problem. The Sustainable Livelihoods framework (Department for International Development 1999) is a widely used framework that reflects this perspective. One dimension of sustainable Development is a concern with "capacity building," which concentrates on developing the skills and capabilities to sustain external initiatives after initial funding is withdrawn. Another approach is "capacity centric," which focuses on sustaining initiatives with funding that builds on those capacities that are present in the community or on finding solutions that capitalizes on the communities' existing capabilities.

A recent interpretation of Development that is gaining much support in the new millennium was articulated by economist Amartya Sen. His book *Development as Freedom* (1999) provides an introduction to this so-called capabilities approach. Sen argues that Development should result in people having greater freedom to make and act on choices about the kind of life they want to live. Of course, lack of finance is one limitation on people's freedom, but there are many others. Sen discusses the importance of other liberating factors such as political freedoms (such as freedom of speech and democratic governance), social opportunities (such as education and social mobility), guarantees of transparency (from agents of government and other wielders of power), protective security (health care and other social safety nets), and the economic freedom in the form of opportunities and abilities to earn or create a livelihood. The creation of the Human Development Index (HDI) by Haq in the 1990s (Haq 1995) shifted metrics from national income to people-centered policies [23]. The HDI is a measure of life expectancy at birth, adult literacy, combined gross enrollment in education, and gross domestic product per capita. The underlying conceptual framework was inspired by Sen's capabilities model.

Another key influence on HCI4D was the creation of the agenda-setting Millennium Development Goals for the turn of the millennium. In the late 1990s, in part as a result of Development" loans and the fact that only a handful of "developed" countries had met their promise at the UN to raise aid to 0.7% of their GDP, critics pointed out that many countries in the "developing" world were paying more back to the developed world in interest on past loans than they were receiving in aid from donors. International campaigns such as Jubilee 2000 exerted significant pressure on political leaders to act (Carrasco, McClellan, and Ro 2007). The Millennium Development Goals agreed at the UN set out eight specific targets that the world aimed to achieve by 2015 (UNDP 2010). The targets focus on poverty and hunger, education, gender equality, child mortality, maternal mortality, HIV/AIDS, environmental sustainability, and building a global partnership for Development. This framing of Development provides a broader vision of Development that focuses on many features that are important for quality of life.

The recent growth of ICT4D can be traced to the 1998 World Development report, which highlighted the role of information, knowledge, and ICTs in Development (Heeks 2009). The G8 Digital Opportunities Task Force in 2000 set up an agenda for action on ICT4D. Following which, the World Summits on the Information Society was held in Geneva in 2003 and Tunis in 2005. Heeks describes a shift in ICTD from late 1990s-early 2000s to the late 2000s model of ICTD 2.0, marked by a shift in goals from realizing Millennium Development Goals to social Development and growth (Heeks 2009). He also notes a significant departure from the telecenter/Personal Computer approach to mobile phone-based solutions. Moreover, Heeks identifies a shift in key Development actors, from philanthropic organizations and donors funding NGOs to "South-based" funding from governments and private players.

62.2 CHALLENGES

HCI4D takes a user-centered approach to both design and Development. The role people from the local communities play is key and may include co-designing, using software in context, co-evaluating the project, and/or reflecting on their own role in the process. Good HCI (in research and in practice) requires an understanding of context and users, iterative design, opportunities for testing and evaluation, and ultimately uptake and application of technologies. UCD always requires that we are keenly aware of not only our participants and their context but also of differences between our own expectations and assumptions and those of our participants. The cultural diversity of “developing” societies has important implications as far as UCD is concerned. For example, under normal circumstances in “developed” regions, user and task analysis techniques offered by HCI provide adequate information about users and their work. However, these techniques are inadequate when a large number of cultural variables must be factored in. To cope with cultural diversity and still ensure optimum performance, a designer needs to know about a much wider and variable range of factors that will affect a person’s work and social behavior. This implies that the emerging HCI practitioner cannot function effectively without including ethnographic techniques.

62.2.1 Local Conditions

Local realities in the “developing” world pose particular challenges that HCI researchers in “developed” regions do not typically face. These can include massive poverty, weak social capital, significant inequality of access, power, and wealth, and weak governance and resulting corruption. Legal structures after often very different from those encountered in “developed” contexts, as are local economic and regulatory environments. The political, economic, and international climate can introduce instability and even danger. While many sustainability issues are relevant in both developed and emerging economies, the way they play out with respect to the users frequently differs.

These same local characteristics manifest in problems with infrastructure that can affect working conditions for local people and the practitioner/researcher. For instance, electricity is often limited or absent, even in urban areas, at certain times of the day or year. Communications can be extremely difficult, and ubiquitous cell phone service is often not available, although it is improving in most geographies. Transportation infrastructures are often variable and can range from quite good to quite bad in the same country at different times of the year and/or in different regions.

In light of these systemic differences, it is not surprising that users and their context typically differ from the people who HCI professionals encounter in “developed” regions. They may lack skills often taken for granted, such as reading, writing, and/or counting in their primary language, although they may be able to speak fluently in multiple languages. They typically lack physical access to technology and, therefore, may have very different experience with and skills to use technology. When technology does exist, it is often shared or may be used in a “mediated” way, where one more technically “savvy” person handles technical use by others with less experience. Technology that may be available may not be the most appropriate—for instance, some parts of “developing” regions are “dumping grounds” for older technology from more “developed” regions and/or locally relevant applications may not be available.

Users’ expectations can be very high—because of a belief in the transformative power of technology—or very low as a result of experience with or exposure to previous technology failures. Similarly, trust in technology can fluctuate widely depending on previous experience and on-the-ground realities.

Therefore, when working with communities in “developing” regions, researchers and practitioners must pay particular attention to the ways in which these local conditions can influence their own work. While some of these factors are similar in any setting, others can be intensified or magnified by the challenges of infrastructure and user characteristics, making it particularly important for the HCI4D researcher/practitioner to deal with them proactively whenever working in contexts that are significantly different from their own. The risk of misunderstandings, assuming too much, and starting from an irreconcilable position exists in virtually any cross-cultural work, but especially when work takes place across a developmental “divide.”

62.2.2 Impetus for Development Work

It is important that we, as researchers and practitioners, examine our comportment, processes, and intentions carefully before undertaking this kind of work. The sensitivities mentioned earlier throw a spotlight on our practices and increase the potential for misunderstanding. There are two key elements to starting out: we need a clear idea of what we are hoping to offer and to gain and we need to understand ourselves well enough that whatever happens in the field, we can respond with flexibility and integrity. In this context, understanding oneself goes beyond the individual and extends to a person’s capacity to understand the place and a way of life they represent. If we are working for a famous corporation, we are often seen as its emissary even if we do not feel like we are important in that corporation. If we hail from somewhere known to be a rich country, although we may not feel affluent, we have to understand that we may be perceived that way. We may come from a country that used to run the administration in the place we are visiting. It may make us seem authoritative even though we wish to be seen as open to ideas.

Nonetheless, it is critical to any design intervention to elicit the local idea of Development—what does the target community consider as empowerment or progress? Juxtaposing these perspectives, that is, the local idea of Development with that of the researcher’s, leads to interesting and useful tensions.

One key difference in outlook is likely to be over how participants in the work see the benefit of being involved.

Local people who become involved in researching or trialing an idea may expect the concept to be developed fully and brought back for use. But, in academia, there is often no intention to see the object of the research into a product through to the point where is it useful to the people with whom it is being researched. And in industry, any products that are inspired from the research are not likely to appear for several years.

Part of understanding what we have to offer and to gain involves understanding and being clear about the benefits that will come from conducting our work successfully in the field. Quite often, there is no immediate benefit to the group who is helping us with our work, despite the long-term potential of our learning. For us, on the other hand, there may be a series of professional accolades: As researchers, we may be able to generate publishable research findings and disseminate our work. As practitioners, we may be able to capitalize on our discoveries by launching new products or opening new markets. The learning may pass from the department that is seeking to understand and design for the communities involved to the people who are wondering whether there is a way of commercializing the learning. Therefore, we have to recognize the potential for repeating a pattern of exploitation. A careful consideration of how a local community will benefit from our research needs to be made. Since HCI4D has the social and moral intention to improve lives of people, each project should be designed in such a way that it yields immediate benefits regardless of its long-term potential benefits. For example, research carried out by Sukumaran et al. (2010) concerned the evaluation of the role of intermediaries between information from a computer and the subsistence farmer who required information. The research was carried in rural India in a remote region of substance farming. Since there was no immediate benefit for members from the local community to participate, the researchers designed the experiment around a service to have the farmers' soil analyzed and subsequently inform the farmers about what types of fertilizer would be best for the crops they intended to grow. Even though this is a very particular example, it illustrates that it is possible to ensure that local communities always benefit from participating in research.

Like any project, at least part of the true benefit of HCI4D work is expected to be realized in the future, when the innovative ideas developed during the project are implemented and scaled. To expect or strive for immediate outcomes would be a misallocation of effort—understanding, experimentation, analysis, and innovation are important despite the seemingly much more pressing needs on the ground. But certain ethical questions are particularly important to pose, especially as we grow our understanding of HCI4D and what this entails. Norms are still being established. Seeing that they develop with continual and deliberate consideration of the intended beneficiaries is an important responsibility of this emerging field. Are we taking more than we leave? Have we respected the priorities found in the field? Are we representing the interests that we found fairly to the rest of the world? The answers to these questions determine the extent to which the incentives of HCI4D align with the ultimate goal of improving the lot of the world's poorest people.

For those who are not in the academic community, the question of motivation may seem to be moot. After all, practitioners are often working in the context of an organization/company, which is interested in expanding a market, and understanding people in that new market is a critical step in that process. However, these HCI4D practitioners often find themselves in situations where what they learn "on the ground" is radically different from what they or their company had expected and where they have to find ways to bridge between their organizations' goals and what they discover. And no one is exempt from considering how the expectations of the local participants on the ground marry up with those of the incoming team and managing those expectations appropriately.

62.2.3 Collaborative Work with Partners and Informants

HCI4D is concerned with the interests of people in some particular location even if there are many different locations in a project. Here we talk about ways of thinking and acting that deal with those local details. If the project involves multiple sites, it will be sensitivity to these details that makes each location study valuable and useful. For instance, Light and Anderson (2009) focus on differences in context and process between rural Chile and rural India in discussing how to develop a global tool for supporting micro-enterprises. But, whether single or multi-sited, the people involved may come from diverse sources and that in and of itself creates interesting dynamics. Even a single study located in one spot may involve partner agencies at that location; it may include people based at another location in the same country; it may include people from other countries who live locally for a few months or years; and/or people whose home is outside the country who perhaps visit occasionally or not at all.

A team of international researchers and practitioners may represent a relatively expensive resource within a project, especially if their work includes a lot of international travel. Models of research and practice that maximize local contributions may be preferable. However, it is not always the case that international collaboration is a negative for an HCI4D effort. International colleagues sometimes add credibility to a project and local contributors can gain status from working with them. In addition, the very real cross-cultural perspectives that all team members bring in to such situations are extremely powerful, especially when consciously used in integrative ways.

One mistake that is easily avoided is to assume that HCI specialists working in research institutions or major corporations based in the country in which you are working will know all about the conditions in other areas of their own country. They may, if they have been working there, but they may never have explored these aspects closely. (Many of us have done little Development-style research in areas of urban or rural deprivation in our own land. We would have to learn about the distinct qualities that make these places what they are.) Social divides may be even greater in "developing" countries, where access to education is more patchy and economic distribution more extreme. So, while a project team with local staff is valuable for the shared learning, capacity building, and more, it cannot be assumed that the people in a university working on HCI know the same as people in the NGO down the road just because they are all from one area and project partners need to be assembled with this in mind.

Teams engaged in HCI4D may face competition between the goals of capacity building, reliability of research, or the quality of design and engineering. As always, there are tradeoffs to be weighed. Having researchers from local universities participate in the work and/or involving local software developers to contribute can aid capacity building but may involve working with larger numbers of less experienced people, requiring a different skill set than is needed in HCI practice or research in other settings. This could also affect the reliability of the research or the quality of design and engineering. On the other hand, such capacity building may contribute to sustainability and have long-term impacts on future Development in the region.

Choices about decision-making authority, responsibility for different aspects of the work, budgets, and how accountability will be distributed between the different stakeholders, including the accountability of the project team to the community, will have a major impact on all the different quality dimensions of the work. A strong, experienced community-based organization (CBO) or NGO may be able to ensure that potential risks of harm to community interests are minimized and benefits maximized.

Being in both an international and interdisciplinary field, HCI4D researchers may find themselves looking for design opportunities in a completely foreign context and domain. Therefore, considerable effort must be put into familiarizing themselves in the new space before beginning the iterative design process. This can include multiple rounds of field work using methodologies like contextual inquiry, qualitative interviews, surveys, field observations, etc. A common strategy is for a researcher to work closely with a grassroots NGO or CBO. The staff at such organizations usually have years of field experience and are very familiar with the daily lives and practices of intended users of any innovation. They can serve as local informants, as well as facilitate meetings, interviews, and observations, enabling an insider role for the researcher. A caveat to this, however, is that if researchers are perceived as associates of the partner organization, they might inherit any negative traits as well. In addition, it is important for the researchers to learn to differentiate opinions of the organization from those of their target users as well, which can be done through triangulation with multiple sources.

Some researchers take the approach of designing a solution that directly aids the partner organization. For example, a researcher might design a tool to simplify an NGO's process of health data collection and analysis and study how this improves the NGO's ability to pinpoint and address health problems in the field. Focusing on strengthening the capacity of an NGO can help ensure that human resources will be available to sustain the use of the technology. However, the flip side is that technologies might end up catering too specifically to the needs of one particular organization, making the result difficult to replicate in other places. Therefore, some researchers engage with a partner NGO mostly as a means of entry into a village, but then work directly with people in the community or village to identify more scalable solutions to widely prevalent problems. For example, the researcher might design a way for villagers across a nation to share video messages about political issues with one another, as well as with higher government authorities.

Researchers from outside the village coming into a village often find themselves in a position in which the villagers attempt to give the answer which they hope will please the researcher the most. This is especially the case with researchers possessing technology (Cheng, Ernesto, and Truong 2008). However, this also holds true in general, and in some cultures it will be considered impolite to say no to the guest (Anokwa et al. 2009). So, it is not unusual for an interviewee to hide their opinion to provide more *socially acceptable* responses. Simply showing up in a village in a car can give off the impression that the researcher has strong political or financial power, influencing the dynamic between the researcher and village residents. The presence of technology gadgets often draws a lot of excitement and interest initially, but one must keep in mind that it might eventually wear off. Many of these challenges can be overcome if the researcher spends extensive time in the field with the mindset of making friends, rather than recruiting subjects. The more the researchers can adapt to conditions in the field, by sitting and sleeping on the floor, eating the same food, or traveling by foot rather than by car, the less "foreign" he or she becomes. While all this takes time, it can make interactions more meaningful and informative.

62.2.4 Internal Conflicts

It is frequently necessary to collaborate with one or more partnering organizations or departments to conduct HCI4D research. These different groups often have very different goals and methods. It is rarely the case, for instance, that an aid-based organization has the same goals, time scale, or budget as the researcher. It is incumbent upon the researcher to listen to and accommodate the needs of the partnering organization(s); however, it is also important to select the appropriate organization(s) based on the researcher's own requirements.

Anokwa et al. (2009) relate a number of stories in which partners are disappointed by broken promises, suggesting from experience that it is important to manage expectations effectively. How does one decide who to design with and what impact to have? Light and Anderson (2009) talk about the shifting sands that beset the research team that has money and motivation but whose outcomes will be largely determined by which set of stakeholders will be involved in the study. How do you choose whose agenda to prioritize or which subset of potential beneficiaries to involve as informants to the design process? Clearly, these questions can be resolved where there is enough time budgeted for a thorough investigation of the wider socio-technical context, but the diversity of issues to consider often makes this a lengthy consultation and deliberation process.

HCI4D industrial projects have a different set of internal conflicts. Practitioners are often well versed in negotiating priorities for research and projects with different groups within their organization, reflecting the different needs and requirements of different functions. However, this becomes particularly important when an HCI4D project is being conducted to understand a new market or region. Unlike other projects, international projects often face the tendency to overload the research with many diverse agendas in the interest of "getting the most" out of the research investment. This risk is greatly heightened with HCI4D projects, especially when they are a company's first experience with a new region. It is critical to negotiate the priorities carefully and to be very clear on what "success" would look like.

In summary, there are a diverse set of people to manage in HCI4D contexts and a greater number of perspectives, but good leadership and management practices still apply. Where there are conflicts, some means of resolution is necessary and, though the mechanism may vary according to the situation (and there will be cultural differences in the acceptability of a more decisive vs. participative approach, for instance), patience, attention to detail, and respect for the reasons that viewpoints differ have and will make these circumstances into opportunities for learning and deeper working relationships.

62.2.5 Scoping

The scope of HCI4D projects needs to be carefully defined, since their viability is often limited by funding and/or by time. HCI4D often involves situations where the scale of the challenge is immense and the budget of time, money, and person power insubstantial in comparison. Can poverty alleviation in an urban slum be actualized by tracing the problem to a single root cause, such as unemployment (hence, job generation as a solution)? Or, is it related to broader issues, such as overpopulation, lack of education, and social inequalities? Establishing causal relationships may help in defining scope for design, possibly creating positive impact. Defining the time frame of the project also helps in designing relevant solutions. Projects need to be designed with sustainability—cultural, environmental, and economic—in mind. Development often involves working with several parties, such as NGOs and donor agencies. Contemplating the future of the project when the designer leaves the field is crucial in the solution design and should be considered from the start.

For example, a project by Sambasivan et al. in an urban slum, the high rate of alcohol abuse was disturbing—women were harassed by their husbands on an everyday basis and family budgets were significantly reduced by alcohol expenses (Sambasivan, Cutrell, and Toyama 2010). The researchers were tempted to create a design intervention to counter domestic violence. However, they felt they were ill-equipped to address a deep socio-cultural issue in their short-term intervention, possibly at the risk of exacerbating existing arrangements without understanding cultural mechanisms. They, therefore, focused their attention on healthcare and education—two important and relevant Developmental areas that generated positive responses and avoided a serious, controversial issue. The ambitions of the project matched the resources.*

62.2.6 Environmental Constraints

More than usual, HCI4D projects take place in a context that offers real restraints to design solutions. Several practical factors affect what is possible in designing and in terms of what we design, such as designing amidst disruptions, cost constraints, recourse constraints, heterogeneous illiteracies (textual, numeric, and symbolic), and communal norms (Spicker no date).

In most developing countries, a divide exists between the technological "haves" and "have nots." Most users in developing countries have access to only very specific technologies such as radio, television, and mobile phones. There may be a sharp urban/rural divide, with slums and shanty towns flanking the more affluent parts of a town or city. Each area will have very different social protocols, access to resources, and expectations. For instance, rural areas in the "developing" world, especially those defined as "deep rural," often experience the following realities:

- Literacy is low, especially among women, and female participation in the public sphere is limited.
- Settlements are scattered spatially and for many residents quality healthcare, agriculture information, and formal education are out of reach or expensive to access.
- Distances to services and facilities are long and often roads are poor and severely affected during rainy seasons.
- Transport services are infrequent in places, which further constrains the accessibility of local residents.

These conditions severely limit the ability of residents to access basic services, social infrastructure and support, attain higher education, or have regular social interaction (social networks) to name a few.

* It is worth bearing in mind too that domestic violence has been seen to increase as a result of certain kinds of take-up associated with ICT use, such as women successfully running their own phone companies (Bantebya Kyomuhendo 2009). While not every eventuality can be anticipated or planned for, thinking about the potential implications of different kinds of intervention allows us to conduct work that has a meaningful outcome for all concerned and avoids leaving a legacy of distrust or destabilization when we depart.

However, such challenges can provide fertile stimuli for innovation in design. In their work on rural microfinance services, Parikh and Lazowska (2006) designed architecture ["CAM"] for mobile phones that would improve efficiency by decreasing the time taken between transporting paper documents between self-help groups and regional offices and decrease error rates in documentation. The CAM framework operates locally on mobile phones, capturing bar codes using the phone camera. It supports existing paper-based interactions familiar to the microfinance operators and minimal navigation, being suited to infrastructural and user constraints. Medhi et al. (2006, 2008) created text-free user interfaces for non-literate users, which included semi-abstracted hand-drawn images for easy comprehension and audio prompts for constant help.

62.3 PRACTICAL GUIDANCE IN GETTING STARTED ON AN HCI4D RESEARCH PROJECT

62.3.1 Before Commencing Fieldwork

As mentioned earlier, the cross-cultural collaboration literature is replete with vital information that should inform HCI4D research and projects. See Anokwa et al. (2009) for more comprehensive coverage.

The effort expended in the early parts of a project on establishing mutual respect and trust between researchers and the local community can be critical (International Development Research Centre 2005). To build trust, early communication has to be a two-way process. Researchers need to listen carefully to understand community concerns and priorities and not simply focus on their external project goals. This demands attention to cultural and nonverbal cues, which may be unfamiliar. It may be valuable at this stage to attend events and meetings organized by the community or by project partners. Acting as a helpful outsider with a willingness to assist others in reaching their goals can also help establish the researchers' credentials as co-operative partners. Only when such trusting relationships are established will it be possible for an external researcher to obtain genuinely valid feedback and data from research participants or to effectively facilitate the identification of problems and priorities. As with any project, the relationships between researchers and other stakeholders will require continuous attention throughout the project.

Following the initial relationship-building phase, projects should define a clear agreement between the organizational partners that clarifies what their different roles will be. The form of such agreements may depend on the capabilities and customs of the collaborating organizations. It may be a Memorandum of Understanding (MoU), a more formal Project Initiation Document, or in a few cases, it could even be a legally binding contract. The point of developing such a document is not that the research group will seek to impose the contract terms. However, the activity of discussing such a document ensures that key issues are made explicit and agreed at an early stage and that project partners integrate the project into their own set of priorities. Key issues include how the activities will be funded, what responsibilities each party has to ensure for the sustainability of any intervention, and how the ethical conduct of research will be monitored and ensured. An additional benefit of such a document is that it can be used as a reference during later project planning and can be particularly useful if there are changes to key staff in the project partner organization. The MoU can then be used to brief new staff to ensure continuity of support. Being sensitive to local cultures and customs of showing commitment will improve relationships with local communities involved. For instance, in some cultural settings, it may be valuable for a senior member of the team to visit the project during this initiation phase to demonstrate the commitment of the "hidden" external partner.

After this, as with other field research, there is usually an introductory session to formally kick off the project's research stage. Often this is a good moment to set expectations of each other and mitigate misinterpretations. Honesty and humility can go a long way in establishing transparency and trust. Inherently, there may be power differences between the researchers and study participants (social class, income, language, skin color, and so on). As standard field practice, it helps to level difference by wearing traditional clothes, sharing meals or sweetmeats, sitting on the floor, or adopting other culturally relevant norms, and above all, revealing genuine concern (Sambasivan et al. 2009). For instance, it may also be appropriate to provide informants with gifts—usually utilitarian ones such as talk time top-ups, utensils, or bed spreads, or symbolic ones, such as certificates of training, photo prints, or sweets. Placement of gift-giving is important so as not to affect responses dramatically. Gifts should be proportional to local incomes (Sambasivan et al. 2009).

62.3.2 Agreeing on Ownership and Responsibility

Another issue to consider early and begin to discuss with other partners is the answer to the question "what is going to happen when the money runs out?" Most academic research is funded for a limited time, and the end of the funded period needs to be a consideration in all major decisions. External academic HCI4D researchers should plan carefully for their withdrawal. This planning should influence research design decisions, as the sustainability of any system after funding is withdrawn will be dependent on the capabilities that are already available or have been developed locally during the project.

Key questions to consider are the following:

- Who within the community or organization will own or control any equipment when the project comes to an end?
- Who will manage any services?
- How will equipment and software be maintained or replaced when required?

- What relationship will the HCI4D researcher/practitioner maintain with the community or organization?
- How will the transition and handover be managed?
- How might the level of support be "tapered" so that the community or organization has time to learn to manage without external support?
- What other resources (local or external) might help to support the work?

62.3.3 Planning to Evaluate Research Results

The other thing that is best considered at outset is evaluation. Evaluating the outcomes of any Development activity is always complex for various reasons. Key challenges lie in the possibility that successful Development involves dynamic social change, so that situations may be changing during the evaluation period; that data capture may be complicated by the social, cultural, and language distinctions between external evaluators and project participants; that ethical considerations may rule out certain forms of data capture and analysis; and that Development involves multiple stakeholders each with legitimate, but different, evaluation criteria.

In this situation it is important that evaluation (and more generally project monitoring) are discussed and continuously reviewed from the earliest planning phases of the project. Many major donors provide useful guides to different approaches to monitoring and evaluation (see, e.g., World Bank 2004; Kasturiaracchi et al. 2009).

Evaluation of HCI4D-related issues needs to be contextualized within the broader context of the community's goals and the agreed concept of Development operating in the project. The link between observable interaction activities and Development project outcomes is rarely (if ever) straightforward. HCI4D evaluators may face choices between evaluating attributes that are easy to measure, but may have little direct relation to Development outcomes (e.g., ease of use or ease of learning of a technology), and attempting to evaluate complex impacts on Development goals, which may be less easily attributable to the specific HCI decisions. These choices are then added to the usual concerns for internal, external, and ecological validity. Typically, a mix of different evaluation questions posed at different levels will be necessary. Thus, it is vital that questions of what to evaluate, how to evaluate, and planning towards such evaluation are addressed at the very beginning of projects and subject to constant review.

62.3.4 Doing Research and Projects

HCI4D Researchers and practitioners working in developing regions sometimes find it difficult to use traditional HCI methodologies. Indeed, many early HCI4D studies focus on appropriate methodologies for developing regions, rather than novel designs or implications for design. However, as mentioned earlier, methodological challenges are not unique to work in developing regions—it is clear that many challenges arise from the fact that researchers are working in cross-cultural contexts and in an interdisciplinary field. Challenges more specific to HCI4D research include language and localization barriers, societal barriers (e.g., political institutions, governments), potentially low prior exposure to technology/training, control over participant selection, and geographical limitations.

It is important to be able to communicate with key informants, whether the researcher or practitioner is performing experiments, observing behaviors, or conducting interviews. Some researchers spend time in formal language training prior to working in a particular country, as a means of overcoming language and localization barriers but, while this might be desirable, it is almost never practical for researchers working in multiple countries, or in countries with many spoken and written languages, or for most researchers in industry. Workarounds include working with populations that speak English (generally more wealthy, educated) or working with and through translators. However, even with translators or English speakers, it is necessary to be careful with interview instrument construction and with preparing the translators. Interview questions must also be constructed carefully from the cross-cultural and cross-technical context: often concepts used in HCI projects like "information" or "data" are not well understood, and particular phrases or gestures may carry different meanings in different contexts. For more unstructured fieldwork, it is important to spend time with translators to make sure they deeply understand the purpose of the research as well as the specific terms or concepts that are important to probe on or observe. Well-trained translators are vital members of the team and can become very adept at noticing subtleties that might otherwise remain hidden.

Often, we are also unable to carry out our research or projects in the manner we expect due to expectations or limitations of the society in which we are working. Anokwa et al. (2009) note experiences in which it is necessary to ask the government for permission before conducting research or projects in a particular country. Braa, Ola Hodne, Johan (2004) also describe how participatory methods are less effective in Cuba than in South Africa. In some countries for instance, it is not appropriate for a man to interview women or vice versa. International team members are often judged by a different set of rules than local team members. Therefore, in some locales, an international woman could interview a local man without violating cultural norms, but a local woman could not. In other places, the social class, ethnic group, or caste of the local team members can open (or close) doors for the team. International team members may make it easier (or harder) for the team to get access to the "right" participants. It is very important to be aware of these kinds of factors that can impact our projects and research, to plan accordingly, and to monitor the situation for possible impacts: being ready to make changes as appropriate to gather the best information we can in order to have the best impact.

In remote and technologically disadvantaged areas, technology poses a constantly changing general methodological challenge. From the researcher perspective, working in villages with no or unreliable electricity makes it hard to depend on audio recorders and digital cameras. It is necessary to carry lots of backup batteries, data cards, and even solar chargers. Introducing users to new designs or new devices also poses issues. Because of low prior exposure to technology, more time must be allocated to training and familiarization, and the impact of the novelty of the technology is important to factor in.

Geography plays a strong role in the typical HCI4D study, since it is hard to return to the investigation site, 2- to 4-week trips are the norm, limiting the depth of human observation to relatively short-term studies. In some cases, researchers are able to return to the same sites multiple times or to stay for longer periods. In these cases, results from earlier co-design or iterative Development cycles are redeployed over multiple trips, with development occurring between experiments. Although time consuming, these types of studies seem to have mixed but generally informative success (Ramachandran et al. 2010a; Luk 2009).

HCI4D practitioners are often even more seriously constrained time-wise than academic researchers. Trips are rarely more than 2 weeks per country and often focus on more urban populations because rural visits are harder to arrange and travel to. Therefore, practitioners more often focus on spending a relatively short time (up to 1 day) with a number of people individually or in naturally occurring groups such as families, communities, or groups of friends. This typically precludes developing the kinds of personal relationships with participants that academic researchers may be able to form. However, this is not altogether a hindrance. It is possible to collect rich information when working closely with local researchers and a team that includes community members and helps identify critical issues in the context of that specific locale. Such a cross-cultural team is particularly critical in this type of focused industrial research and can provide far more nuanced, robust, and useful insights than any monocultural team can hope for. “Insider” and “outsider” participants both bring significant perspectives.

62.4 CASE STUDIES

The challenges in preparing for HCI4D projects, in identifying and assuring commitment from partners, involving local knowledge and communities, distributing and sharing responsibilities, setting goals and expectations, carrying out the research, measuring success, and capitalizing on local culture are each highlighted in the following six case studies. The case studies cover different types of HCI4D projects, in a variety of international cultural settings, and report different problems and solutions. Together, they offer a set of possible approaches toward successful HCI4D.

62.4.1 First Days Project Case Study

(Divya Ramachandran)

62.4.1.1 Overview

Across developing regions, women are still dying due to preventable complications in pregnancy and childbirth, a problem almost entirely eradicated in the industrialized world. The First Days project aims to address an aspect of this issue by providing a method for delivering clear, persuasive messages about pregnancy and delivery care for mothers-to-be. Specifically, the project targets rural health workers and helps them to establish credibility in their communities as health resources, empowering them to effectively convince pregnant women and their families to use maternal health services despite conflicts with traditional customs. The First Days project is being carried out in the state of Orissa in eastern India. Orissa is typical of a number of economically disadvantaged states in India, which struggle with health outcomes in spite of a number of government and non-government health efforts.

To improve maternal health in India, the central government has established a National Rural Health Mission, which employs one woman from each village to serve as an Accredited Social Health Activist or ASHA. ASHAs are charged with promoting free government health services (like subsidies for institutional deliveries, immunizations, and prenatal care) and providing counseling on pregnancy care, family planning, breastfeeding, and so on. In over 2 years of qualitative research with ASHAs, pregnant women, their families, and other key community players, we learned that ASHAs are employed in most rural villages in India but their training, effectiveness, and acceptance within the village is still minimal. Traditional beliefs and rigid social structures limit the change that they promote.

62.4.1.2 Understanding Needs

We identified a number of barriers to information uptake and dissemination in the communities. First, the structure for continuing education of ASHAs does not support the acquisition of new knowledge and information. ASHAs attend a monthly sector meeting where they are trained on one new topic each time. In a particular meeting we attended, roughly half of the ASHAs employed in the sector were present. Moreover, the teaching method (formal lectures on a general topic) was not conducive to the ASHAs’ education level and schooling experience. Experienced trainers of health workers from successful NGOs suggest that rather than formal lectures, ASHAs would benefit more from information that is relevant to specific situations that their clients are in as well as instructions on how to effectively share this information with their clients. Second, we found that ASHAs had not received training in maintaining an effective work routine. For instance, client monitoring and counseling were aspects unfamiliar to the ASHAs. As a result, the local community did not find the ASHAs’ valuable resources and the limited information they could offer was rarely accepted.

Given these challenges faced by ASHAs, we concluded that they could be empowered by more effective tools to share health information with their clients. Considering the educational, language, and cultural barriers they needed to overcome, videos on mobile phones would be a valuable tool to communicate to clients what they needed to do in specific health-related situations. The mobile phone platform was portable, prevalent, and appropriate for ad hoc counseling visits to clients and their families. We felt videos could convey information comprehensively and consistently in an engaging way for low-literate audiences. Furthermore, as ASHAs shared videos with their clients, they themselves would have the opportunity to learn both factual information and effective counseling techniques over time.

We created two types of videos: testimonial and persuasive (Ramachandran et al. 2010a). Testimonial videos provided social proof of the ASHA's role and importance. We trained ASHAs to record videos using their cell phones and asked them to record messages from influential individuals in their own villages. Their videos featured village leaders, pregnant clients, and even street plays addressing topics including endorsements of the ASHA's role and importance, their own personal health experiences, and instructional messages. These videos provided persuasive social proof to clients that other villagers believed and followed the messages promoted by ASHAs. Persuasive videos were designed as short informative segments about pregnancy-related health issues, which we selected based on prevalence in the target communities. We identified relevant content from health handbooks (such as Hesperian Press publications) and adapted these to health advice in line with local resources and cultural practices with the help of local nurses. A local artist sketched some basic illustrations for the content, and we asked staff at our partner NGO to record voiceovers. We strung each video together with panning and zooming to give it an animated feel. Each video lasted between 30 seconds and 1 minute.

We deployed these videos with seven ASHAs for 2 months and we observed ASHAs using these videos on house visits. Our initial observation was that the mobile videos provided ASHAs with a concrete example of the information that needed to be brought across to the client and thus helped them understand what to accomplish during house visits. We also observed an increase in knowledge and self-efficacy by ASHAs. However, ASHAs had to be trained extensively to use the videos. For instance, we encouraged them to pause the videos, discuss the topics depicted, and engage their clients in conversations about the videos. However, during our observations, we did not find that all ASHAs adopted the technique of engaging their clients by pausing the video at relevant moments and asking related questions to their clients spontaneously. We had to provide explicit training about how to use the videos in this way to ensure the videos were regularly used by the ASHAs, and the effectiveness of the videos depended largely on each individual ASHA's ability to grasp this task of pausing and discussion. Therefore, the videos did not seem to ensure any consistency in the quality of counseling. This experience spurred us to create a second version of the videos with two important differences. The new version had a built-in dialogue to facilitate client engagement regardless of whether the ASHA was comfortable with this counseling style. The background voiceover asked questions that required yes/no input; the videos automatically paused at these points to facilitate responses and discussion. The second innovation was the use of persuasive message architecture. We identified prevalent myths and barriers that stopped women from performing particular behaviors and addressed these directly by providing corrections and solutions, respectively. For example, anemia is a serious problem among rural women and is the direct cause of one-fifth of all maternal deaths. ASHAs distribute free iron tablets to all pregnant women, but many women believe that taking these tablets causes the baby to get too big to deliver normally. A typical lecture-style video modeled off of textual information might present an explanation of iron-deficiency and the risks of anemia and then suggest some actions such as improving diet and taking iron tablets. However, our dialogic video opened directly with a question, "Do you believe that taking iron tablets will cause your baby to get too big, leading to complications during delivery?" This was then followed with a correction, "That is not true; in fact, iron tablets will give you more strength to get through a normal delivery," and so on. The value of such a dialogic approach was that women immediately related to the topic of the video and began to discuss this widely believed myth with either the ASHA or others present, and each expressed her opinions on whether it was true. This increased relevance of the message to the client empowered the ASHA to communicate more effectively.

We found that the quality of the counseling session was significantly improved when ASHAs used the new versions of the videos. ASHAs spent more time discussing various aspects of the message and also elaborated on the message more frequently. From our observations, we found that clients showed more interest and were more attentive when the health worker used the phone messages compared with providing the information orally without the use of videos on the mobile phone (Ramachandran et al. 2010b).

62.4.1.3 Lessons Learned

62.4.1.3.1 Lesson 1: Enabling Access to Information Is Not Always Enough

Our first design attempted to capture important health messages by using mobile videos, which we believed would be both informative for health workers and engaging for their clients. Yet we observed that health workers still had little idea of how to use the videos to engage their clients, and clients had difficulty finding the videos personally relevant. Therefore, it was critical for us to create dialogic videos that guided the health workers through a conversation with their clients and incorporated persuasive techniques to engage clients and help them see the relevance of the messages. While access to health information appears to be the premise of this project, the focus on motivation of health workers and persuasion of clients is critical. This becomes obvious only through a very careful and in-depth understanding of the local context of maternal health care.

62.4.1.3.2 Lesson 2: Consider Contextual Factors in Research Design

Although our results show that dialogic messages significantly improved the ASHAs' ability to provide in-depth, effective counseling, we were unable to measure the impact of this improved ability reliably. Throughout the process of designing a formal study to evaluate the impact of the messages, we made changes, iterations, and compromises to our research goals in response to real, challenging factors in our study environment. This is often a disadvantage in field studies, which attempt to maximize realism and, consequently, have limited control over extraneous factors. We learned the importance of considering these factors not just when we studied the context, but through the research and evaluation process. It is important to consider the tradeoffs of prioritizing methodological rigor over accurately preserving the interactions of contextual realities.

In our case, the rurality of our selected field site limited our ability to recruit more participants and introduced some confounding variables. For example, we found that the ASHAs varied individually in their persuasive power, and a possible cause could have been their caste. We found that ASHAs of the same caste, as most of the client base, were more persuasive than those from higher castes. While we cannot make strong conclusions from our limited sample size, this serves as a reminder that various social, cultural, or political factors can be tightly coupled with the success or failure of ICTD projects.

62.4.2 Big Board Case Study

(Gary Marsden, Andrew Maunder, and Richard Harper)

62.4.2.1 Overview

The Big Board project's goal is to build a system that could download information to any handset for free. The idea for the system came from ethnographic studies and interviews with rural mobile phone users in Zambia and Malawi. Many people in these areas have a handset but cannot afford airtime to make calls or send messages. Most people interviewed were hoping that someone would call them or send them a message. At the same time, we were working with various NGOs that needed a way to distribute information (e.g., on HIV/AIDS, voting materials, etc.) to this same user group. The NGO professionals were frustrated as the people they were trying to reach could not afford to download the information.

A detailed description of the system can be found in Marsden, Maunder, and Parker (2008), but essentially it uses a large screen driven by a PC media server. Users take a picture of anything they see on the screen that they are interested in and Bluetooth that to the PC server. The server then recognizes what they took a picture of and sends back, via Bluetooth, all the media it has relating to that particular topic.

62.4.2.2 Understanding Needs

Our previous attempts at using *participatory techniques* to design similar systems had proven less than successful—users felt awkward sketching interfaces; they were unsure of what technology could do and seemed to second guess what answer the researchers were looking for rather than expressing their own opinion about what the system should do. Therefore, we took a *technology probe* approach and built a high-level prototype to deploy within the community. By presenting the community with a functioning system, users could interact with it and start to suggest usages for the system. To manage and record these interactions, we identified a Human Access Point (HAP) within the community, who was paid to record reactions to the system and interview people about its potential uses. We have found the identification of a HAP to be advantageous in a number of projects; by choosing a person from a target community who has spent some time working with technology (in this case, the HAP had taken a course in Windows and Microsoft Office), they are able to translate between their community and that of the researchers. For the purposes of this project, we coached the HAP in how to keep a diary of usage of the system, which she updated daily. We also encouraged her to interview people who were using the system about what they were doing and why. Clearly, this approach is open to biasing, which is why we triangulated the diary results with usage logs and the polyphonic evaluation reported below.

Building on the feedback the HAP provided, we were able to refine the prototype and better fit it to the needs of the community, much as any iterative design process would. To evaluate the system, however, we could not rely on the HAP as she was personally involved with the success of the project. Instead, we turned to *the polyphonic assessment* techniques of Gaver (2007). Much like the work in the developing world, Gaver seeks to create systems outside the users' previous understanding of what technology might be capable of. To provide an unbiased opinion of the system, Gaver employs journalists to interview users and elicit opinion; on the basis of that journalists are highly trained in gaining opinion and to be unbiased. In our project, we recruited two journalists who were from the same language group as our deployment community. The journalists were merely told that some technology had been deployed in the community and we wanted to understand peoples' reactions and thoughts on how it had impacted them. The journalists recorded opinions and peoples' reported usage patterns, which we were then able to triangulate with usage logs recorded by the system. The back stories provided by the journalists were key in understanding usage patterns and tailoring of content. For example, one participant reported that she downloaded media to consume with her husband in the evening—she found it had greatly improved their relationship. A mother used the system-downloaded media simply to show her children that she could get media on her handset, just as they could.

62.4.2.3 Lessons Learned

We identified several issues in the methods we have described earlier. These issues highlight how these methods differ from creating systems in the developed world context: First, in the developing world, it is often unclear what the problem is and much ethnography has to be conducted to explore the nature of the problem space. In this instance, the need to build a system grew purely from studies with rural users and NGOs—at no point did anyone explicitly ask for an information sharing system. The need for such a system grew from the observation that many people had media-capable handsets but lacked the funds to download media from the internet—one person we interviewed had been listening to the same three preloaded songs on his handset for over 12 months! Second, the iterative design process was not conducted with end users, but a proxy for these users (the HAP). Although this is not an ideal process (an ideal process would be where the community were able to build their own technologies), we have found it provides better results than working directly with end users; we refer to this process as "mediated design." We are actively researching the types of compromise we introduce by using a proxy for users, rather than the users themselves, but are convinced that the interpretations the HAP provides are more culturally appropriate than we had been inferring from direct user studies. Finally, for many developing world projects, standard evaluation measures such as effectiveness and efficiency only represent a part of the evaluation; we believe it is critical to go beyond that and measure how the introduction of the technology impacts quality of life. In this example, we chose to evaluate the system according to its usage patterns and through understanding how users consumed the information it provided, which led us to Gaver's polyphonic assessment. In other projects, we have used measures of domestification (Silverstone and Haddon 1996) or appropriation (Heeks 2005) to gauge success. In projects where there are NGO partners, we have even used their success measures to assess the impact of our work.

So, while HCI contains many methods for the creation and evaluation of system designs, our experience has been that these methods cannot be applied "blindly." In other words, we are often driven back to the original assumptions that drove their creation to reinterpret them for users living in the developing world. Not only does this stretch HCI to cover issues in the developing world, but we expose holes in the methods that the rest of the world is using, which can strength practice in the developed world as well.

62.4.3 Rural e-Services Case Study

(Andy Dearden)

62.4.3.1 Overview

The Rural e-Services project combined three methodological questions: The first was how participatory methods used in development could be combined with the methods and traditions of participatory design; The second was how to develop a platform to offer multiple e-Services in rural areas by making use of mobile devices; Third was how to create sustainable business models in challenging environments. The project was concerned with rural areas in India, and we had initially been funded on that basis that we would begin by exploring microfinance. However, as we engaged more deeply with the community and the community-based organizations (in our case, working through an NGO that had both microfinance and agricultural initiatives), we discovered that community members and leaders viewed improving "agricultural information flow" as a higher priority than microfinance development.

62.4.3.2 Understanding Needs

We followed a "deeply embedded" strategy for participation (Dearden and Rizvi 2009). This began with a 3-month period of "entry to the field" (International Development Research Centre 2005), which involved general discussions, getting to know the community, and building relationships. During this period, as well as listening carefully to understand the interests of the community, the researcher was also able to show commitment to the community, for example, one's helping some community members to get home from the state capital when they had run out of funds due to travel delays. This period culminated in a project establishment meeting between the project, the Sironj Crop Producers Company Ltd. (a co-operative of marginal farmers), PRADAN (an NGO), and the state governments District Poverty Intervention Program, where the overall direction of the project was revised to develop an improved "agricultural information flow system." The software design was driven using a combination of techniques from participatory IT design, participatory development practice, and agile software development. These included timelines to explore history and important sequences of activities, chapatti diagrams (which are used to discuss priorities with size reflecting importance; see Figure 62.1), storytelling workshops, scenario generation, and short software development cycles (about 1 month for each cycle).

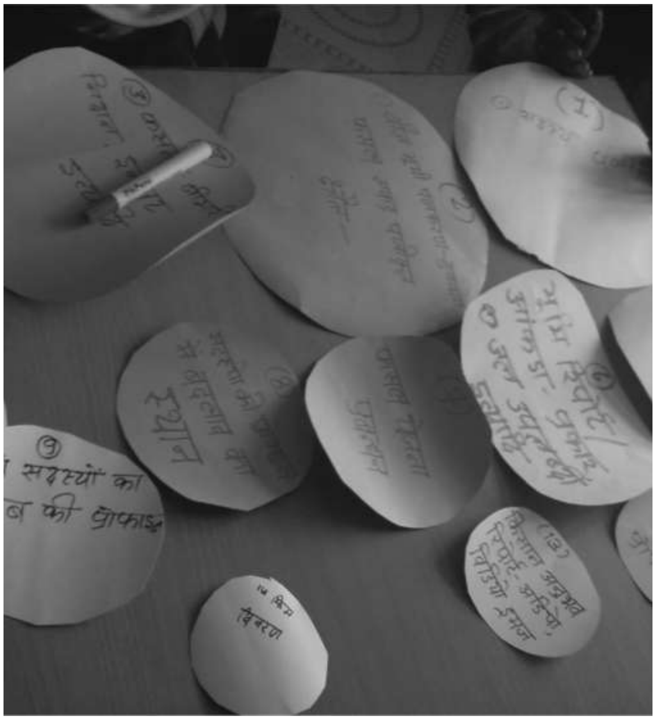


FIGURE 62.1 **(See color insert.)** Example of a chapatti diagram.

The software was developed by Safal Solutions, a software company from Secunderabad, Andhra Pradesh, specializing in systems for the NGO and development sector. Although we could not arrange for the farmers and the software developers to be co-located, during each cycle a delegation of farmers travelled to Secunderabad for 2 days to test and suggest revisions to the designs.

The software produced was called Kheti (which stands for Knowledge Help Extension Technology Initiative and means agriculture in Hindi). The main component of Kheti is a mobile phone-based application (built in Python on a Symbian 60 phone), which allows farmers to create simple multimedia messages that consist of up to six photos plus an audio track. This is then sent to a web server. The CEO and agricultural advisor of the farmers' co-operative can then view these messages (usually they did this in the evening) and call back to the service provider and farmer the following day for a discussion. Other components were a web-based membership database for the co-op so that the advisor knew what the farmers were growing and what their land holdings were and an Interactive Voice Responder System that could store recordings of advisory conversations. The system was field-trialed for 3 months with over 200 multimedia messages being exchanged, and some major pests and diseases were avoided. However, we did not succeed in finding a sustainable business model that could pay all the costs involved in running the system.

62.4.3.3 Lessons Learned

The primary lesson was that participation in design is not primarily about the methods that are used such as paper-prototyping (see Dearden and Rizvi 2008a,b for a discussion), but is about how designers, researchers, and beneficiaries interact and work together. Participation implies negotiating control of not just the design of the technology, but also of goals of the project, the processes of designing the definition of success. For example, we had to engage all the stakeholders from the funders to the farmers to negotiate the switch from microfinance to agricultural information, the software design and development schedule had to be adapted to fit with the seasonal nature of the farmers' work, and the conceptions of success had to recognize organizational development of the co-operative as an important outcome in its own right. During the project, many of the co-operative members gained from considerable personal development, particularly gaining the confidence to interact with and assert their understandings and their interests when dealing with the software developers, government officials, the researchers, and the NGO. The researchers and software developers also learned and were changed by our experiences.

Another important lesson was that effecting change through HCI4D is highly dependent on the input of partner organizations. Kheti can be understood as an information system for the co-operative, and as with other information systems, its introduction has to be led by internal champions. One of the challenges for HCI4D is to organize projects so that the community hosting and operating the technologies take on ownership and drive the project for themselves.

An interesting finding for us as HCI researchers was that when people see real value in a system, they will invest a lot of time and energy learning how to use the technology. In our case, we would have liked to improve the user interface, but project timescales and limits on funding meant that we had to move to field trials before we were completely happy. However, we discovered that the end users were willing to learn to use the system because they could see the potential benefits for themselves.

We began with an assumption that using agile software methods and rapid prototyping would be beneficial to create an interface that was well matched to our users. In practice, we found that we had to allow extra time in each iteration for users to learn and understand the elements of functionality that were already available. However, software prototyping was important so that our users could understand the degree to which software is malleable and can be revised.

Finally, we found that all the farmers we worked with were able to make meaningful inputs to design decision making, but the NGO staff and more educated farmers were better equipped to respond to partial design concepts and ideas about how technology could be. For example, when we tried to develop some future-oriented scenarios and storyboards, many of the farmers were unable to understand what we were asking them to do. Farmers were more comfortable telling stories about their current lives and past experiences, but were unfamiliar with this new kind of storytelling.

62.4.4 Post Conflict Computing Case Study: Video Sharing and Reconciliation in Liberia

(Thomas N. Smyth and Michael L. Best)

62.4.4.1 Overview

In 2003, a comprehensive peace agreement finally brought an end to almost two decades of brutal civil war in Liberia, a small West-African nation of about 3.5 million residents. In the wake of that disastrous conflict, Liberia, like many nations before it, established a Truth and Reconciliation Commission (TRC) charged with investigating the causes and effects of the war and helped to establish a lasting peace. The Commission was the first of its kind to truly embrace new digital media in the service of its goals. As part of the Commission's media strategy, our research group was tasked with building a mobile computer video story-sharing kiosk through which citizens could share their thoughts and opinions on the war and the nation's future path. In undertaking this project, we encountered a heretofore unexplored field for computer interaction design and Development: that of post-conflict reconciliation. This case study recounts some of the novel challenges we faced and the HCI innovations we developed to meet them.

62.4.4.2 Understanding Needs

Based in Atlanta, Georgia, our team faced many practical obstacles in working directly with the Liberian end-users using traditional collaborative design processes. In response, we developed a methodological innovation called HDF, Heuristic → Diaspora → Field (Best et al. 2008). Under this methodology, candidate designs were first reviewed in our laboratory by local usability experts based on widely accepted heuristics. Next, designs were tested with members of the considerable Liberian diaspora (some 10,000 members strong) in Atlanta. We performed user studies and focus groups at various locations including a Liberian restaurant, which also served to expose our Atlanta-based designers to a local, ready-made Liberian "culture capsule" (Foucault, Russell, and Bell 2004). Finally, we shipped designs refined through this process to Liberia for the critical stage of end-user field testing. The interim step of working with the diaspora proved particularly fruitful, leading to early changes in design direction including an emphasis on extreme simplicity, along with helpful feedback on the appropriateness of our choice of symbols, icons, and voice prompts.

Out of this three-part design process emerged the Mobile Story Exchange System (MOSES) (Best et al. 2009; Smyth, Etherton, and Best 2010). The video-sharing concept behind MOSES is simple: citizens approach the kiosk, browse and watch video messages created by others, and record messages of their own. Rather than relying on extremely scarce Internet connectivity for dissemination of content, all videos remain physically within the kiosk, which is itself transported throughout the country.

MOSES is pictured in Figure 62.2. It consists of a screen, a set of ruggedized buttons, a camera, directional microphone, and speakers. Chassis fans and air filters keep the inside of the system cool and dust-free. A glare shade allows viewing outdoors, where MOSES is usually deployed. The system is powered using marine deep-cycle batteries and can run for up to 8 hours completely untethered. All components are housed in a Liberian-made lockable and sturdy wooden housing, allowing the system to be left unattended for short periods. Finally, the system can be separated into two halves for easy transport.

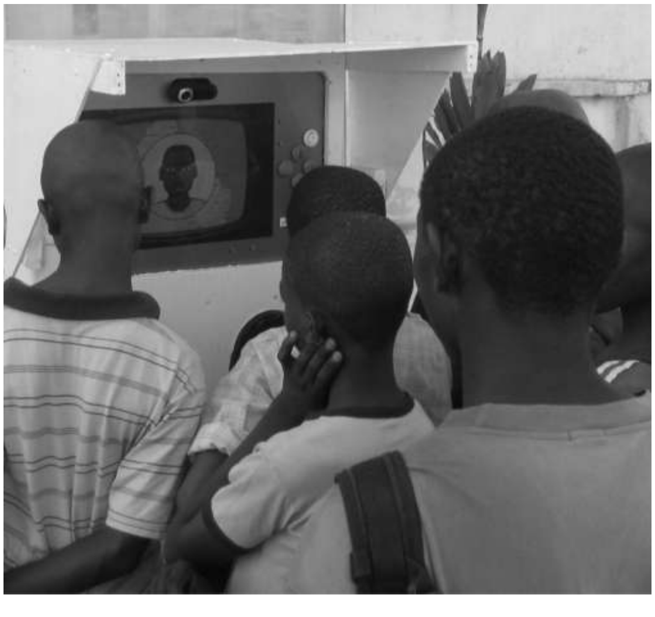


FIGURE 62.2 (See color insert.) People using the Mobile Story Exchange System (MOSES) kiosk.

The user interface features Moses, an embodied conversational cartoon agent, also shown in Figure 62.2.

Moses guides users through browsing, watching, and recording processes using simple verbal prompts such as "Please look into the camera and remember to speak loudly. Press the white button when you are ready." This design allows walk-up use of the system by Liberians with no computer or print literacy. While embodied conversational agents have been widely panned within the HCI community, we find some possible rehabilitation to this design approach when targeting illiterate user populations. The purpose of MOSES is to support public discourse about Liberia's past, present, and future. The public nature of the system is made clear to users by Moses' voice prompts as well as by the physical locations of the system, usually in highly public places surrounded by a crowd of people.

62.4.4.3 Lessons Learned

Over a period of approximately 2 years, MOSES was carried to most major areas of Liberia and was experienced by thousands of Liberians. Over 900 videos were recorded and saved. During this period, we also carried out two evaluations of the system, one qualitative and one quantitative. The results from both suggest that interactive video story sharing systems can indeed help in post-conflict national healing and reconciliation. One demonstrative quote came from a participant who recorded a video about unjust employment practices:

> [My video] will go places and people will witness it, and they will know what happening in some area. For them to know what is happening they will get to know it is not happening in one place, it is happening all over.

Given the complete novelty of the video content creation and sharing technology to most participants, we found this strong and immediate embrace of the medium remarkable. In addition we found that Moses, the animated conversational guide, was warmly received by participants who spoke of him as a "friend" and "teacher" and reported relying heavily on his instructions. Given floundering interest in animated conversational agents in most user interface design circles, the strong enthusiasm expressed by our study participants was intriguing. Finally, we observed that MOSES was almost always used by a group of people rather than by a single individual. Many participants reported that they drew inspiration, confidence, and technical assistance from the group.

General self-efficacy is believed to be important when establishing conditions for sustaining post-conflict peace. If we can demonstrate that interacting with our system can increase a participants' generalized self-efficacy, then we will have to take a critical step towards establishing our broader hypothesis that rich digital media is an important tool in post-conflict reconciliation (Long and Brecke 2003; Ropers 2004). With intrastate conflict emerging as the predominant form of armed conflict in the modern world, it appears that the need for effective new approaches in post-conflict reconciliation is needed. We believe that appropriately designed ICTs can form part of such strategies, and MOSES has been an encouraging first step in that direction. We are proud and grateful to have had the opportunity to work with the TRC of Liberia, and we are currently investigating other potential projects in Uganda and Afghanistan.

62.4.5 Low-Cost Media and Mobile Phones in Data Elicitation: Designing a Method for HCI4D

(Nithya Sambasivan)

62.4.5.1 Overview

A central challenge to conducting research in low-income contexts of developing countries is designing or tailoring methods for data gathering and evaluation. Particularly, in situ studies, which are crucial in understanding development in the real world, are especially difficult to orchestrate. Several interesting challenges may affect the application of traditional HCI methods in such contexts—low literacy rates, technology skill and usage differentials, and social and economic stratifications, in addition to ambient noise, electrical power, and cost constraints. Western-world techniques do not map easily to such contexts, because of profound differences in users, needs, contexts, practices, and goals of projects.

In the spring of 2009, we conducted an ethnographically inspired study in two urban slums of India (Nakalbandi and Ragigudda) to understand low-income communities and their engagements with technology (Sambasivan 2009a,b). We primarily employed ethnographic techniques, such as participant observation, household surveys, and semi-structured interviews, to understand the socioeconomic, developmental, and cultural aspects of our informants. By triangulating our data with inputs from the communities and the NGOs, we determined two key developmental areas that were of interest and relevance to the community—education and health care. To help understand the reality of technology and social structures of the communities, as well as to pilot some technology created for the above-mentioned development areas, we designed *ViralVCD*, a low-cost, rapid data elicitation method (Sambasivan, Cutrell, Toyamo 2010). The technique leverages local practices and existing infrastructure to elicit contextual data. It employs physical media and mobile phone questionnaires to gain access to data on multiple levels: *social* networks underpinning information diffusion; *technological* ownership, access, and usage; and *developmental* impact assessment of HCI4D projects. ViralVCD is an example of a larger class of possibilities that can be seen as a methodological contribution to researchers working in resource-challenged contexts.

62.4.5.2 Understanding Needs

Our initial ethnography pointed to the relatively high penetration of televisions, video compact disc (VCD) players, and mobile phones. Associated with these technologies were the practice of "missed calls" (terminating a phone call before the receiver picks up, to cut costs) and the prominent role of entertainment in everyday life.

To produce useful and interesting content for the pilot, a participatory video framework was created. Based on our ethnography, we highlighted local best practices, along with expert advice, in the format of VCD videos. While the videos provided us content for social development, they also served as a lens to study their own diffusion in the communities (i.e., understand how videos get viewed, by whom, where, why, etc.).

Based on our initial findings, we created videos around the areas of education and healthcare.

62.4.5.2.1 Education

Parental lack of literacy was attributed to poor academic performance of children and high incidence of dropouts, in our study. Based on our ethnography, we elicited best practices in teaching among certain parents whose children enjoyed academic success. Inspired by the heavy viewership of soap operas in these communities, we scripted a role-play between two non-literate women, demonstrating the best practices. Techniques to ensure good academic performance that overcame the non-literacy barrier were demonstrated, for example, making children read aloud and looking for ticks and crosses. An education expert provided actionable steps.

62.4.5.2.2 Healthcare

Our informants attributed their poor nutrition to their low income, and they would often fall sick from eating unhealthy food. We hosted a "cooking contest" in the slums to extract local knowledge in an openly competitive fashion. Taste and nutritional value were used as judging criteria to motivate healthy cooking. Snippets of the contest were embedded into the final video. This was followed by a segment on balanced diet.

62.4.5.2.3 Dissemination

As a next step, we burned the videos (roughly 12 minutes each) onto VCD, owing to their low costs and ubiquity. Then, we screened the videos in six slums using the local VCD player and television. Inspired by the participatory format in Digital Green by Gandhi et al. (Gandhi et al. 2007), where videos featured local members of the community, our hypothesis was that people may want to view the videos more because their peers from similar communities feature in them. Following the screenings, we handed out VCDs with unique identifier numbers and celebrity photos attached to the sleeves (Figure 62.3). At the end of each video, a visual and auditory prompt provided instructions to dial a "missed call" to the phone number on screen (also written on the VCD), which was ours. We immediately called the number back to conduct a short interview, mainly to enlist

FIGURE 62.3 **(See color insert.)** (a) Sample video compact disc and sleeve. (b) A screening session in Jakkur.

the caller's socioeconomic profile and to gauge their understanding and feedback on the content. If the caller answered a content-related question correctly, they were provided with a utilitarian prize (utensil or blanket) as a gift. They were then encouraged to pass on the VCDs to others in their social circle. The contest was limited to a week, after which, calls were no longer accepted.

We distributed 132 VCDs to 65 attendees. For these attendees, the call response rate was 31.25%, with 20 callers. In total, 50 unique callers were registered and 31 VCDs were transferred.

ViralVCD helped us generate social, developmental, and technological insights. Key social insights include tracing paths of transmission (by mapping out who passed VCDs to whom) and the social processes driving the diffusion. At a microlevel, two forms of diffusion emerged—the prominent, peer-to-peer propagation (A→B→C) and actor-driven diffusion (A→ (B and C)). Peer-to-peer propagation was seen in communities where multiple key (active) actors existed, and actor-driven diffusion was visible where there was a strong actor with a strong social network. At a macrolevel, the diffusion reflected the social solidarity of the community—neighborhoods splintered by heavy internal politics showed fewer proclivities towards diffusion activity. Tightly knit communities exhibited quick and widespread responses.

62.4.5.3 Lessons Learned

Key technological insights include understanding the communal usage of technologies: the place, time, and nature and composition of the group in which the shared activity transpired; the working order of VCD players, televisions, and mobile phones; and the correlation between technology ownership and communal participation. Finally, ViralVCD helped in creating developmental extensions for education and health. Because we placed contest details at the end of the video and asked unique questions, viewers needed to watch the entire video to answer correctly. We queried on the understanding and usefulness of the content in health practices and child rearing.

ViralVCD was helpful in identifying critical agents in communities, in understanding their socio-technical makeup, and in identifying and recruiting peers of the same socioeconomic stratum through snowballing. ViralVCD avoided additional infrastructure in understanding community capital, technological ownership and access, and developmental baselines. It complemented our ethnography by providing understandings of organic use, users, and contexts of use, which could be applied to the design of HCI projects.

62.4.6 MILLEE Case Study: Mobile and Immersive Learning for Literacy in Emerging Economies

(Matthew Kam)

62.4.6.1 Overview

The MILLEE (Mobile and Immersive Learning for Literacy in Emerging Economies) research project aims to improve "power language" literacy among low-income children in developing countries. MILLEE revolves around the idea that immersive, engaging, and yet educational games on cell phones that target language literacy can make high-quality learning more accessible to low-income children in underdeveloped regions who lack access to high quality schooling.

Low literacy is without doubt one of the grand challenges in the developing world. But even more challenging is the tension between regional languages and global "power languages," such that economic opportunities are often open to those literate in a power language. For instance, even though more than 20 regional languages are spoken widely in India, English language is widely perceived to be a socioeconomic enabler. The economists Munshi and Rosenzweig (2006) estimate that English speakers in Mumbai experience returns on investment in schooling that are between 24% and 27%. On the other hand, non-English speakers with similar characteristics experience returns that are about 10%. English is thus the language of power in India, such that mastery of the language can almost be associated with membership in the middle and upper classes in India (Faust and Nagar 2001; Kishwar 2005).

Unfortunately, the public school systems in developing regions such as India have poor outcomes. According to a literature review commissioned by the Azim Premji Foundation (2004b), public schooling is out of reach for more than 43% of school-going age children in rural areas who cannot attend school regularly due to their need to work for the family in the agricultural fields or households.

On the other hand, cell phones are increasingly adopted in developing regions to the extent that the cell phone has become the fastest growing technology platform in the developing world (Vodafone 2005). We believe we can dramatically expand the reach of language and literacy learning in the developing world by using portable mobile devices such as the cell phone as the target platform, so as to enable children with work commitment to access educational resources anytime and anywhere, at places and times that are more convenient than school alone.

62.4.6.2 Understanding Needs

At the time of writing, the MILLEE team has made more than 10 trips to India to conduct field research, including iterative design and pilots studies, since the project started in 2004. These trips were also crucial for building relationships with local partners and communities, whose support were instrumental for the success of our pilot studies. In such a multidisciplinary endeavor, the MILLEE team comprises computer scientists, HCI specialists, second language and reading acquisition specialists, as well as videogame designers. The team includes members who, through their experiences growing up in India and/or volunteering with humanitarian organizations working to improve education there, possess a deep knowledge of the local cultural context.

The games we have designed and developed to date in this new suite of MILLEE games collectively targets one semester of English as a Second Language (ESL) curriculum as mandated by the state government of Andhra Pradesh in India.

In the case of the MILLEE project, the designs of our e-learning games draw on best practices in the state-of-the-art language learning software (Kam et al. 2007a), as well as a cross-cultural analysis of the qualitative differences between traditional Indian village games and contemporary Western videogames (Kam et al. 2009). Our goals are threefold: (1) to design pedagogical applications that are informed by the research base on reading literacy and second language acquisition; (2) to take the best practices in commercial language learning applications as a starting point as opposed to reinventing the wheel; and (3) to design educational games that are culturally consistent with the traditional village games that constitute the play experiences of rural Indian children.

62.4.6.3 Lessons Learned

Design knowledge for instructional design: In terms of language pedagogy, we situate our technology and instructional design processes within the Task-Based Language Teaching (TBLT) curriculum development framework (Ellis 2003, Nunan 2004, Prabhu 1987, Skehan 1998). In TBLT, the learner engages with a series of pedagogic tasks that take the form of goal-directed activities.

Our rationale for using TBLT as a guiding framework is twofold. First, the task has an inherent degree of structure that designers can follow and fits well with the existing work practices of designers and educators: just as the language instructor can plan her teaching tasks prior to her classroom lessons, the instructional designer can devise tasks for computer-assisted learning systems that are eventually deployed with learners. Second, TBLT has demonstrated learning outcomes in the Indian context. Prabhu (1987) describes the well-known Bangalore/Madras Communicational Teaching Project, in which task-based approaches for teaching ESL in India were experimented with children aged between 8 and 13 years over a 5-year period. In an independent evaluation, Beretta and Davies (1985) found that children taught using TBLT performed significantly better than their counterparts in the control group on tests of transfer that evaluated them on contextualized grammar, dictation, and comprehension (both listening and reading).

The challenge, however, is that theoretical frameworks such as TBLT remain too abstract for technology designers to grasp easily, much less translate into software designs. At first glance, it appears that the design process calls for multidisciplinary collaboration between designers and educators, in addition to specialists from other domains such as game design. Such a partnership is no doubt necessary but insufficient. In our experience, the typical language teacher may not know how to draw on her teaching experiences and background, so as to imagine concrete designs.

On the other hand, there are existing commercial language learning products that include games and other software. We take the state of the art in existing commercial language learning products as the initial basis for our instructional design and avoid reinventing the wheel. Along this line, how can we capture the existing design knowledge reflected in the instructional design of current language learning software and commercial products?

The formalism that we use for capturing this design knowledge is the design pattern (Alexander 1977), which grew out of building architecture and is increasingly popular in domains such as software engineering. The MILLEE project is arguably the first attempt to apply design patterns to the domain of language pedagogy. A design pattern provides insights into a frequently encountered design problem by describing the problem, the essence of the solution to the problem, the rationale for the solution, how to apply the solution, some of the tradeoffs in applying the solution, and related design patterns. A primary benefit of a design pattern is to encourage the reuse of existing solutions to problems that are frequently encountered, so as not to reinvent the wheel. Design patterns, especially those that capture the design knowledge employed in the instructional design and videogame design of successful language learning games, thus constitute a design tool that designers can use to create more of these games while maintaining reasonable educational quality.

In the MILLEE project, our design processes leveraged a set of more than 50 pedagogical design patterns that we distilled from a review of over 35 commercial language learning applications. We conducted this review in a principled manner by using task-based language teaching as our analytical lens. We selected this sample of more than 35 commercial applications based on the following factors, which we adopted as our proxy indicators for educational quality: a large professional customer base, highly educated users (in the case of adult learners) or parents (in the case of children's software), as well as strong reviews and/or ratings on home schooling, e-commerce, and so on, websites. Our sample included ESL learning software packages that are developed specifically for non-English-native low-income students from the rural areas and urban slums in developing regions (e.g., the series of software developed by the Azim Premji Foundation in India for use in over 15,000 affiliated rural Indian primary schools), best-sellers in the foreign language learning market (e.g., *Rosetta Stone*, Simon and Schuster's *Pimsleur*, Topic Entertainment's *Instant Immersion*, and Auralog's *Tell Me More* series), as well as early literacy games (e.g., the Learning Company's *Reader Rabbit* and Scholastic's *Clifford: The Big Red Dog* series).

62.4.6.3.1 Design Knowledge for Gameplay Design

Next, in designing effective e-learning games, gameplay design is an equally important design dimension in the MILLEE project. As such, we have also experimented with design patterns that capture some of the design knowledge in gameplay design, including heuristics for enjoyable gameplay. However, while we have been able to use the above pedagogical design patterns to culminate in positive learning outcomes (Kam et al. 2007a), our results with game design patterns were less successful (Kam et al. 2007b). Rural children did not necessarily find the games whose designs were informed by game design patterns to be necessarily intuitive, exciting or free from playability problems. It appeared that rural children have relatively little exposure to these videogames, whose designs were influenced by Western cultural traditions that unconsciously incorporated into the game design processes. It seemed that patterns require adequate knowledge of the cultural context to be employed effectively.

We conducted contextual interviews with 87 children in villages in North and South India, during which we asked participants to recall the everyday games that they love and to play these games for us to videotape. We observed a total of 23 outdoor and 5 indoor games. Seventeen outdoor games belonged to the family of "tag" games, in which there is generally at least one player designated "it" who has to "tag" players in the opposing team by touching them, either with a hand or an object. In particular, 2 and 3 "tag" games belong to the "cops and robbers" and "hide and seek" sub-families, respectively. The six outdoor games that do not belong to the "tag" family include tug-of-war, kite flying, marbles, hopscotch, and the spinning top. The indoor games can be generally classified as "tabletop" games. To understand what traditional games are made up of, we examined each of the above 28 games and identified the elements (Fullerton 2008; Björk and Holopainen 2005) that comprised their game mechanics. The game elements that we considered included the players, game resources, goals, actions, and rules.

Based on the insights that we have gleaned from the above analysis, we have devised a tool for designing videogames that target children in rural India and potentially other rural regions. Specifically, by providing a detailed description of the elements in traditional village games (Kam et al. 2009), we have provided the community with a "palette" of game elements that game designers can draw from to put together new game designs for children in rural developing regions. As such, this tool is *generative* in that it facilitates new designs. Our preliminary results with this tool suggest that designing videogames with the same game mechanics as those found in traditional games, while leaving out those mechanics that are absent, ensures the most successful videogame designs that rural children can relate to more readily. We encourage other researchers and practitioners who are working on videogames or literacy interventions that aim to improve lives for poor children in the developing world to experiment further with the design tools and knowledge that we have put together in our work.

62.4.6.4 Next Steps

We are extending the latest suite of games to target an entire academic year of English curriculum as mandated by a local state government in India. Each game will focus on an early literacy competency such as phonological awareness, word recognition fluency, or lexical inferencing. Despite the massive undertaking involved in developing digital content that aims to teach an official curriculum, it is necessary for adoption purposes to align our games with an official syllabus so that we can evaluate the efficacy of the MILLEE approach on a syllabus that stakeholders, including parents and government officials, view as an important educational credential. The undertaking is complicated by the need to develop remedial digital content (i.e., "bridge content"), which covers prerequisite knowledge that rural children who previously had low-quality schooling require so that they would be equipped to learn the official curriculum.

The task is further complicated by the limitation that much research on reading acquisition and literacy is based on learners in industrialized countries with reasonably good access to schooling, whereas our target learners reflect vastly different conditions on measures such as "concept of print" and school-based social practices, all of which are important theoretical constructs in existing literacy theories. We have drawn from the existing research base on literacy studies as best as we could to inform our bridge content. We aim to leverage these games as a research infrastructure that can be used to operationalize and test the extent to which existing literacy frameworks apply in—and have to be extended to be more insightful for—culturally divergent environments. In this way, we believe that undertaking HCI in the developing world can help us as a community of researchers and practitioners attain a more complete understanding of what design, cognition, literacy, and learning truly means.

62.5 WAY FORWARD

As this chapter has shown, HCI4D is bringing UCD thinking and approaches to the creation of technology, which aim to support economic and community Development. Of course, we are not alone in trying to bring user-centered approaches to laterally related domains. Others are also trying to do this, including groups promoting sustainable interaction, easy-to-use "Green" technology, human-centered built environments, and usability of voting systems. Like these colleagues, we face challenges of finding new shared ground, learning how to bridge into a new domain, and figuring out how to make our unique contributions valuable in terms of another domain and/or discipline value. However, unlike other efforts toward sustainable and improved quality of life, HCI4D focuses chiefly on users who are not likely to have the disposable income to pay for the technology. The end users' lack of purchasing power is the main distinguishing feature of HCI4D and brings about many of the challenges addressed in this chapter.

We believe that HCI4D can have a positive impact in projects like those presented here. At the same time, we also hope that HCI4D is moving the whole discipline of HCI forward because we are extending, amending, changing methods and tools—and creating new ones—in ways which are not just useful for the "developing" world. While we and others (e.g., Prahalad 2009) believe that the "developing" world is where we see the growth of the technology industry as more and more diverse people become invested in creating and using technology, we also believe that extending our ability to learn about and design for increasingly diverse populations benefits and informs HCI as a whole. Certainly, when we look at what is happening worldwide, it is clear that technology is being appropriated at ever-increasing rates. Local people are building new technologies and applications to meet locally relevant needs more than ever before. That adoption, design, and implementation is happening without necessarily employing the formal principles and techniques of UCD as espoused by the HCI community.

To this point, most of us in the HCI4D community have focused on the notion of "doing good" with the end-goal of improving the quality of life for a local community. We believe that user-centered approaches are inherently "better" for communities as well as for design in that they avoid attempts to impose technologies—that participation "works better" both because it includes previously excluded people and because it results in a better, more sustainable design. This is an admirable motivation. However, we must also find the overlap between what is well-intentioned and what is economically, socially, and environmentally viable and sustainable. Otherwise, our efforts will not scale and will not benefit the wider world. In the worst case, if a technology is not sustainable, it may bring false hopes and disappointment, increasing the digital divide and decreasing the likelihood that future HCI4D projects will succeed. If the outcomes of HCI4D projects are economically viable, it is inevitable that companies and institutions will become involved. It is only with the wider involvement from industry, policy makers, and development actors that HCI4D can truly have impact. In fact, an enterprising mindset is necessary when carrying out HCI4D projects. While economic viability is not a primary concern in most HCI projects, especially HCI research projects, the focus in HCI4D on alleviating poverty inevitably makes economic, social, and political considerations an important cornerstone of HCI4D, and as such, distinguishes HCI4D from much of HCI.

The case studies described in this chapter illustrate this entrepreneurial, innovative spirit. Faced by sometimes debilitating conditions imposed by the local context of where the technology would be used, all the authors discovered new insights to use in innovating and building a more fitting and "smarter" solution. Each of the case studies tells the story of an attempt to deploy a mainstream iterative interaction design cycle and the inventive workarounds and solutions that were found when conventional processes and methods did not carry across. The case studies show that user-centered approaches can be extremely useful; they are necessary but not sufficient. Building relationships with NGOs and local people is a critical first step. All the authors would agree that these relationships are important in the success of their projects. Therefore, we need to reach beyond the academic community, the Development community, and the handful of large multinational companies which have been involved through funding of single efforts, often through their "corporate responsibility" arms. We must include different partners with different constraints, incentives, and vantage points. Otherwise, there is little chance of bringing about the kinds of systemic changes that are required to make UCD, technologies, and programs that scale. We need to have a principled way of thinking about these economic considerations and integrating them into existing HCI frameworks.

Some of those voices may be those of people from NGOs and those of people in local communities, some of whom we already work with and some of whom are as yet unknown to us. They may include local entrepreneurs, local business people, local developers, and people in regional or local companies who are trying to find solutions to "local" problems—solutions that have the potential to be applied more widely with additional resources, including expertise. International business plays a role too. But, like the other players, it cannot end poverty alone. Designing for the market in user-centered ways is, in the opinion of some (Polak 2009, 2010), the only way for affordable solutions to the problem of poverty to measure up to the challenges they meet. "The ruthless pursuit of affordability is an essential component of this design revolution, which in many ways stands on the shoulders of the appropriate technology movement. Most importantly, to be successful, the revolution in design for the other 90% has to develop disciplined ways to design for the market" (Polak 2010). With a project focus on extreme affordability, industry can partner up with NGOs, CBOs, local entrepreneurs, and governments to develop economically viable and sustainable solutions.

Clearly, if we are to help in this revolution, we may find that we need to adopt new ways of working together. Some have argued (Dray 2009, Buie, et al. 2010a,b) that finding ways for academics and practitioners to communicate and work together better is critical for HCI. That is even more critical in HCI4D at the same time that the challenge is greater. Working collaboratively together—just as we seek to work collaboratively with those in local communities—is key to helping HCI4D change the world. There is much to help us with this, for this is a time of great experimentation. Funding agencies like the National Science Foundation (NSF) in the United States are moving away from their traditional levels of grant support, and now NSF "especially welcomes proposals for cooperative projects involving both universities and the private commercial sector" (NSF 2011). Academics are looking increasingly to non-federal sources of funding to support their research and diversify their funding base. Public–private partnerships, originally developed by governments, have been expanded for use in international development (World Economic Forum 2005). As challenging as this is, it is also exciting.

By sharing lessons we are learning as we collaborate with new and different players, as we find ways to incorporate an understanding of sustainability and affordability, as we experiment with adapting time-honored methods and creating new ones that fit an increasingly diverse world, and we are finding new ways of working that can also benefit HCI as a whole. And that will help all of us to change the world in a positive way—to empower, to learn, to create, and to build a better future for all.

REFERENCES

Alexander, C. 1977. *A Pattern Language: Towns, Buildings, Constructions*. New York: Oxford University Press.

Anokwa, Y., T. Smyth, D. Ramachandran, J. Sherwani, Y. Schwartzman, R. Luk, M. Ho, N. Moraveji, and B. DeRenzi. 2009. Stories from the Field: Reflections on HCI4D Experiences. *Inf Technol Int Dev* 5(4). http://itidjournal.org/itid/article/view/427/195 (accessed November 30, 2011).

Avgerou, C., and G. Walsham. 2000. Introduction: IT in Developing Countries. In *Information Technology in Context: Studies from the Perspective of Developing Countries*, ed. C. Avgerou and G. Walsham. Aldershot: Ashgate Publishing Ltd.

Azim Premji Foundation. 2004. *The Social Context of Elementary Education in Rural India*. Bengaluru, India: Azim Premji Foundation.

Bantebya Kyomuhendo, G. 2009. The mobile payphone business: A vehicle for rural women's empowerment in Uganda. In *African Women and ICTs: Investigating Technology, Gender and Empowerment*, ed. I. Buskens and A. Webb. Zed Books & IDRC. http://www.idrc.ca/en/ev-137013-201-1-DO_TOPIC.html (accessed September 26, 2010).

Beretta, A., and A. Davies. 1985. Evaluation of the Bangalore Project. *ELT Journal* 39(2):121–12.

Best, M. L., T. N. Smyth, D. Serrano-Baquero, and J. Etherton. 2009. Designing for and with diaspora: A case study of work for the truth and reconciliation commission of liberia. In *Extended abstracts on Human factors in computing systems*, 2903–18.

Best, M. L., Serrano-Baquero, D., Abbasi, H., Pon, C. L., Roberts, D. L., and Smyth, T. N. 2008. Design of Video-Sharing Kiosks for Liberian Post-Conflict Reconciliation. Unpublished paper presented at *CHI 2008, HCI4D Workshop*. San Jose, CA.

Björk, S., and J. Holopainen. 2005. *Patterns in Game Design*. Hingham, MA: Charles River Media.

Braa, J., T. Ola Hodne, and S. Johan. 2004. Participatory health information systems development in Cuba: The challenge of addressing multiple levels in a centralized setting. In *PDC 2004 - Proceedings of the Eighth Conference on Participatory Design*, ed. A. Clement and P. Van den Besselaar, 53–64. Toronto, Ontario, Canada. http://delivery.acm.org/10.1145/1020000/1011877/p53-braa.pdf?key1=1011877&key2=7878144821&coll=GUIDE&dl=GUIDE&CFID=104403073&CFTOKEN=50873688 (Behind login in ACM Digital Library; accessed September 26, 2010).

Buie, E., S. Dray, K. Instone, J. Jain, G. Lindgaard, and A. Lund. 2010a. How to bring HCI research and practice closer together. Workshop presented at CHI 2010, Atlanta, GA, and included in *Proceedings of the 28th of the International Conference Extended Abstracts on Human Factors in Computing Systems*, New York, NY: ACM.

Buie, E., S. Dray, K. Instone, J. Jain, G. Lindgaard, and A. Lund. 2010b. A. researcher–practitioner interaction. "Special Interest Group" presented at CHI 2010, Atlanta, GA, and included in *Proceedings of the 28th of the International Conference Extended Abstracts on Human Factors in Computing Systems*, New York, NY: ACM.

Carrasco, E., C. McClellan, and J. Ro. 2007. Foreign debt: Forgiveness and repudiation In *The E-Book on International Finance & Development*, ed. E. Carrasco. Published online and available at: http://www.uiowa.edu/ifdebook/ebook2/contents/part4-I.shtml (accessed September 26, 2010).

Chambers, R. 1994. The Origin and Practice of PRA. *World Dev* 22(7):953–69.

Cheng, K. G., F. Ernesto, K. N. Truong. 2008. Participant and Interviewer attitudes toward handheld computers in the context of HIV/AIDS programs in Sub-Saharan Africa. In *The Proceedings of CHI 2008: The ACM Conference on Human Factors in Computing Systems*, 763–66. Florence, Italy. http://khaitruong.com/publications/CHI-2008c.pdf (accessed September 26, 2010).

Dearden, A., and H. Rizvi. 2008a. Adapting participatory and agile software methods to participatory rural development. In *PDC '08: Experiences and Challenges, Proceedings of the Participatory Design Conference*, 221–5. Bloomington, IN: Indiana University Press.

Dearden, A., and H. Rizvi. 2008b. Participatory design and participatory development: A comparative review. In *PDC '08: Experiences and Challenges, Proceedings of the Participatory Design Conference*, 81–91. Bloomington, IN: Indiana University Press.

Dearden, A., and H. Rizvi. 2009. A deeply embedded sociotechnical strategy for designing ICT for development. *J SocioTechnol Knowl Dev* 1(4):52–70.

Department for International Development. 1999. *Sustainable Livelihoods Guidance Sheets*. London. http://www.nssd.net/references/SustLiveli/DFIDapproach.htm (accessed September 26, 2010).

Dray, S. 2009. Engaged scholars, thoughtful practitioners: The interdependence of academics and practitioners in user-centered design and usability. *J Usability Stud* 5(1):1–7. http://www.upassoc.org/upa_publications/jus/2009november/JUS_Dray_Nov2009.pdf (accessed September 26, 2010).

Ellis, R. 2003. *Task-Based Language Teaching and Learning*. New York: Oxford University Press.

Fals-Borda, O. 1987. The application of participatory action-research in Latin America. *Int Sociol* 2(4):329–47.

Faust, D., and R. Nagar. 2001. English Medium Education, Social Fracturing, and the Politics of Development in Postcolonial India. *Economic and Political Weekly* 2878–83.

Foucault, B. E., R. S. Russell, and G. Bell. 2004. Techniques for researching and designing global products in an unstable world: a case study. In *Proceedings of the 23rd International Conference Extended Abstracts on Human Factors in Computing Systems*, New York: ACM.

Fullerton, T. 2008. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. 2nd Edition. Burlington, MA: Elsevier Inc.

Gandhi, R., R. Veeraraghavan, K. Toyama, and V. Ramprasad. 2007. Digital green: Participatory video for agricultural extension. Presented at ICTD '09 and published in Information Technologies and International Development, 5(1):1–15. Available at: http://itidjournal.org/itid/article/view/322/145 (accessed November 30, 2011).

Gaver, W. 2007. Cultural commentators: Non-native interpretations as resources for polyphonic assessment. *Int J Hum Comput Stud* 65(4):292–305.

Haq, M. 1995. *Reflections on Human Development*. New York: Oxford University Press.

Heeks, R. 2005. e-Government as a Carrier of Context. *J Public Pol* 25:51–74.

Heeks, R. 2009. *The ICT4D 2.0 Manifesto: Where Next for ICTs and International Development?* Published online and available at: http://www.sed.manchester.ac.uk/idpm/research/publications/wp/di/documents/di_wp42.pdf (accessed September 26, 2010).

Ho, M., T. Smyth, M. Kam, and A. Dearden. 2009. Human–computer interaction for development: The past, present, and future. *Inf Technol Int Dev* 5(4):1–18. http://itidjournal.org/itid/article/view/420/188 (accessed September 26, 2010).

Humphrey, J. Policy implications of trends in agribusiness value chains. *The European Journal of Development Research* 18(4):572–92.

International Development Research Centre. 2005. *Participatory Research and Development: A Sourcebook*. Vol. 1. Ottawa, Canada: Author. http://www.idrc.ca (accessed September, 26 2010).

Jalava, J., and M. Pohjola. Economic growth in the New Economy: Evidence from advanced economies. *Information Economics and Policy* 14(2):189–210.

Kam, M., A. Kumar, S. Jain, A. Mathur, and J. Canny. 2009b. Improving literacy in rural India: Cellphone games in an after-school program. In *Proceedings of the Information and Communication Technologies and Development 2009 International Conference* (ICTD 2009), April 17–19 2009. Doha, Quatar, 139–49. Available at: http://www.scribd.com/doc/14234869/ICTD-2009-Proceedings.

Kam, M., A. Mathur, A. Kumar, and J. Canny. 2009. Designing digital games for rural children: A study of traditional village games in india. In *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems.* New York: ACM. Available at: http://www.cs.cmu.edu/~mattkam/publications/CHI2009.pdf (accessed November 30, 2011).

Kam, M., D. Ramachandran, V. Devanathan, A. Tewari, and J. Canny. 2007. Localized iterative design for language learning in underdeveloped regions: The PACE framework. *Proceedings of the 25th International Conference Extended Abstracts on Human Factors in Computing Systems*, New York, NY: ACM.

Kam, M., V. Rudraraju, A. Tewari, and J. Canny. 2007b. Mobile gaming with children in rural india: Contextual factors in the use of game design patterns. In *Proceedings of 3rd Digital Games Research Association International Conference (DiGRA '07),* September 24–28, 2007. Tokyo, Japan. Available at: http://www.digra.org/dl/db/07312.25032.pdf (accessed November 30, 2011).

Kasturiaracchi, A., T. Eriksson, S. Rodriques, and A. Kubota. 2009. Planning, monitoring and evaluating for development results. In *Handbook on Planning, Monitoring and Evaluating for Development Results*, 5–17. New York: United Nations Development Programme. http://stone.undp.org/undpweb/eo/evalnet/Handbook2/documents/english/pme-handbook.pdf (accessed September 26, 2010).

Kishwar, M. P. 2005. Deprivations's Real Language. In *The Indian Express*. http://www.indianexpress.com/printerFriendly/12662.html# (accessed September 26, 2010).

Kleine, D., and T. Unwin. 2009. Technological revolution, evolution and new dependencies: What's new about ICT4D? *Third World Q* 30(5):1045–67.

Kraemer, K. L., and J. Dedrick. 2001. Information technology and productivity: Results and policy implications of cross-country studies. In *Information Technology, Productivity, and Economic Growth*, ed. M. Pohjola, 257–79. Oxford, UK: Oxford University Press.

Light, A., and T. Anderson. 2009. Research Project as Boundary Object: negotiating the conceptual design of a tool for International Development. In *Proc ECSCW 2009*, Vienna, Austria.

Long, W., and P. Brecke. 2003. *War and Reconciliation: Reason and Emotion in Conflict Resolution*. Cambridge, MA: MIT Press.

Luk, R., M. Zaharia, M. Ho, B. Levine, and P. M. Aoki. 2009. ICTD for Healthcare in Ghana: Two Parallel Case Studies. In *Proceedings of the 3rd International Conference on Information and Communication Technologies and Development, ICTD '09*, 118–28. Piscataway, NJ. New York: IEEE Press.

Marker, P., K. McNamara, and L. Wallace. 2002. *The Significance of Information and Communication Technologies for Reducing Poverty*. London: Department for International Development. http://www.oecd.org/dac/ictcd/docs/matrixdocs/GBR_paper1.pdf (accessed September 26, 2010).

Marsden, G., A. Maunder, and M. Parker. 2008. People are people, but technology is not technology. *Phil Trans R Soc A* 366(1881): 3795–804. Available at: http://people.cs.uct.ac.za/~gaz/papers/RSTA20080119.pdf (accessed November 30, 2011).

Medhi, I., A. Sagar, and K. Toyama. 2006. *Text-Free User Interfaces for Illiterate and Semi-Literate Users*. Berkeley, CA: IEEE/ACM International Conference on Information and Communication Technologies and Development.

Medhi, I., A. Sagar, and K. Toyama. 2008. Text-free user interfaces for illiterate and semi-literate users. *Inf Technol Int Dev* 4(1):37–50.

Munshi, K., and M. Rosenzweig. 2006. Traditional institutions meet the modern world: Caste, gender, and schooling choice in a globalizing economy. *Am Econ Rev* 96(4):1225–52.

NSF. 2011. Grant Proposal Guide. National Science Foundation. Available at: http://www.nsf.gov/pubs/policydocs/pappguide/nsf11001/gpg_1.jsp (accessed November 30, 2011).

Nunan, D. 2004. *Task-Based Language Teaching*. New York: Cambridge University Press.

Oxford University. 2010. *Department of International Development*. http://www.qeh.ox.ac.uk/ (accessed September 26, 2010).

Parikh, T., and E. Lazowska. 2006. Designing an architecture for delivering mobile information services to the rural developing world. In *International World Wide Web Conference: Proceedings of the 15th International Conference on World Wide Web*, 791–800. Edinburgh, Scotland, NY: ACM Press.

Polak, P. 2009. *Out of Poverty: What Works When Traditional Approaches Fail*. San Francisco, CA: Berrett-Koehler Publishers, Inc.

Polak, P. 2010. *The Death of Appropriate Technology*. Blog post. http://blog.paulpolak.com/?p=376 (accessed September 26, 2010).

Prabhu, N. S. 1987. *Second Language Pedagogy*. Oxford, U.K.: Oxford University Press.

Prahalad, C. 2009. *The Fortune at the Bottom of the Pyramid: Eradicating Poverty through Profits*. Revised and Updated. Wharton School of Publishing/Pearson Education.

Ramachandran, D., J. Canny, P. D. Das, and E. Cutrell. 2010a. Mobile-izing health workers in rural india. In *Proceedings of CHI 2010, Human Factors in Computing Systems*, 1889–98. Atlanta, GA: ACM Press. http://research.microsoft.com/en-us/um/people/cutrell/CHI2010-RamachandranEtal-Mobile-izingHealth.pdf (accessed September 26, 2010).

Ramachandran, D., V. Goswami, and J. Canny. 2010b. Research and reality: Using mobile messages to promote maternal health in rural india. In *Proceedings of ICTD 2010, International Conference on Information and Communication Technologies and Development*. London: Royal Holloway University of London Press.

Ropers, N. 2004. From resolution to transformation: The role of dialogue projects. In *Berghof Handbook for Conlict Transformation*, eds. D. Körppen, B. Schmelzle, and O. Wils. Berlin, Germany: Berghof Research Center for Constructive Conflict Management.

Sambasivan, N., E. Cutrell, and K. Toyama. 2010. ViralVCD: Tracing information-diffusion paths with low cost media in developing communities. In *Conference on Human Factors in Computing Systems: Proceedings of ACM Conference on Human Factors in Computing Systems*. 2607–10. New York: ACM Press.

Sambasivan, N., N. Rangaswamy, E. Cutrell, and B. Nardi. 2009. Ubicomp4D: Interaction and infrastructure for international development: The case of urban indian slums. In *UbiComp: Proceedings of the 11th International Conference on Ubiquitous Computing*. 155–64. New York: ACM Press.

Sambasivan, N., N. Rangaswamy, K. Toyama, and B. Nardi. 2009a. Encountering development ethnographically. *Interactions* 16(6):20–23.

Sambasivan, N., N. Rangaswamy, K. Toyama, and B. Nardi. 2009b. Encountering development ethnographically. *Interactions* 16(6):20–3. http://www.ics.uci.edu/~nsambasi/ACMInteractions09_EncounteringDevEthno.pdf (accessed September 26, 2010).

Schwarzer, R., and M. Jerusalem. 1995. Generalized self-efficacy scale. In Measures in Health Psychology: A User's Portfolio. Causal and Control Beliefs, ed. J. Weinman, S. Wright, and Johnston, 35–7. Windsor, U. K.: NFER-NELSON.

Sen, A. 1999. *Development as Freedom*. Oxford, U.K.: Oxford University Press.

Silverstone, R., and L. Haddon. 1996. Design and the domestication of information and communication technologies: Technical change and everyday life. In *Communication by Design: The Politics of Information and Communication Technologies*, ed. R. Mansell and R. Silverstone. New York: Oxford University Press.

Skehan, P. 1998. Task-based instruction. *Annual Review of Applied Linguistics* 18:268–86.

Smyth, T., J. Etherton, and M. L. Best. 2010. MOSES: Exploring new ground in media and post-conflict reconciliation. In *CHI*. Atlanta, GA. http://mikeb.inta.gatech.edu/uploads/papers/pap0922-smyth.pdf (accessed September 26, 2010).

Spicker, P. no date. *British Social Policy*. 1601–948. Published online and available at: http://www2.rgu.ac.uk/publicpolicy/introduction/historyf.htm (accessed September 26, 2010).

Sukumaran, A., S. Ramlal, E. Ophir, V. RamNaresh Kumar, G. Mishra, V. Evers, V. Balaji, and C. Nass. 2009. Intermediated technology interaction in rural contexts. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems Extended Abstracts* (CHI EA '09), 3817–22. New York: ACM.

The National Science Foundation Proposal Award Policies and Procedures Guide. 2009. Effective January 4, 2010. NSF 10–1, OMB Control Number 3145-0058. http://www.nsf.gov/pubs/policydocs/pappguide/nsf10_1/gpgprint.pdf (accessed September 26, 2010).

Truman, H. S. 1949. *Inaugural Address*. http://www.vlib.us/amdocs/texts/41trum1.htm (accessed November 30, 2011).

United Nations Asian and Pacific Training Centre for Information and Communication Technology for Development (APCICT). 2010. *Briefing Note: ICT Project Management in Theory and Practice*. http://www.unapcict.org/news/aboutus/programmes/research/BriefingNote-7-web.pdf (accessed September 26, 2010).

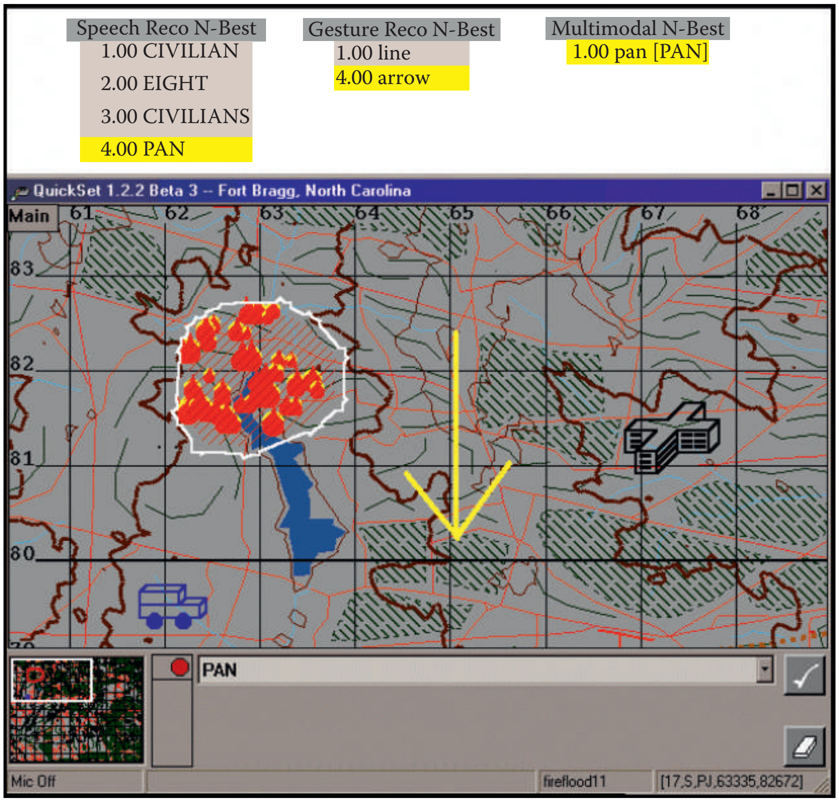
United Nations Development Programme. 2010. *Millennium Development Goals*. http://www.undp.org/mdg/ (accessed September 26, 2010).

United Nations Environment Programme. *TUNZA Network*. http://www.unep.org/tunza/youth/ (accessed September 26, 2010).

Vodafone Group. 2005. Africa: The impact of mobile phones. Moving the debate forward (Vodafone Policy Papers Series No. 2). Available at http://www.vodafone.com/content/dam/vodafone/about/public_policy/policy_papers/public_policy_series_2.pdf (accessed November 30, 2011).

World Bank. 2004. Monitoring & evaluation. Some tools, methods & approaches. *Operations Evaluation Department*. New York, NY: World Bank. http://www.worldbank.org/ieg/ecd/me_tools_and_approaches.html (accessed September 26, 2010).

World Economic Forum, Financing for Development Initiative. 2005. Building on the monterrey consensus: The growing role for public-private partnerships in mobilizing resources for development. In *United Nations High-level Plenary Meeting On Financing for Development*. http://www.weforum.org/pdf/un_final_report.pdf (accessed September 12, 2010).

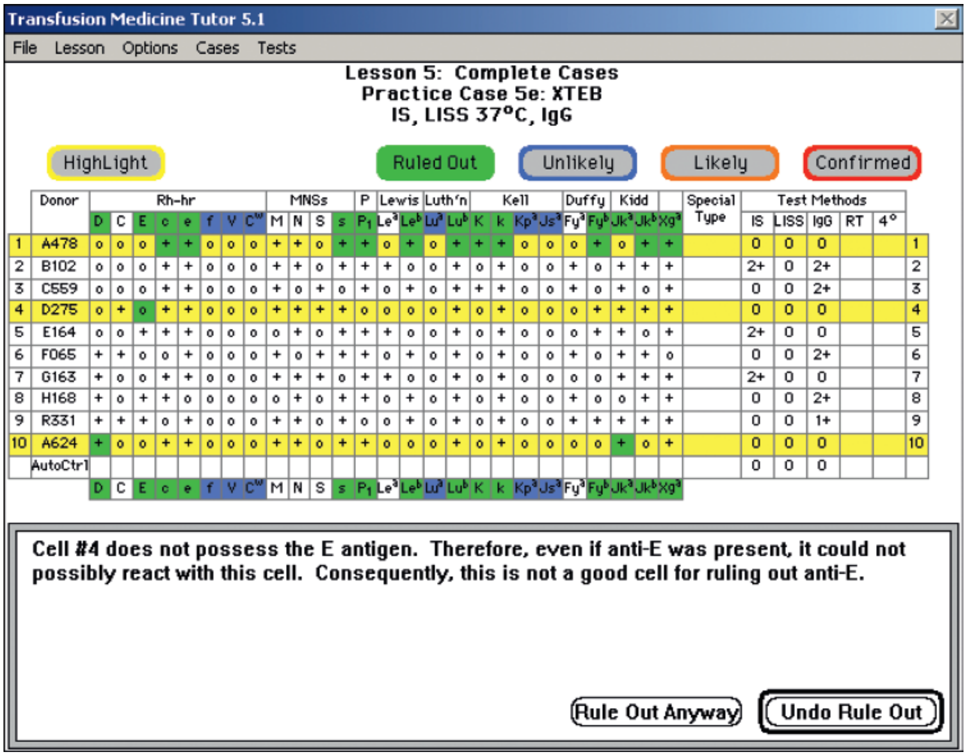


Speech Reco N-Best
1.00 CIVILIAN
2.00 EIGHT
3.00 CIVILIANS
4.00 PAN

Gesture Reco N-Best
1.00 line
4.00 arrow

Multimodal N-Best
1.00 pan [PAN]

FIGURE 18.3 Multimodal command to “pan” the map, which illustrates mutual disambiguation occurring between incoming speech and gesture information, such that lexical hypotheses were pulled up on both n-best lists to produce a correct final multimodal interpretation.



Lesson 5: Complete Cases
Practice Case 5e: XTEB
IS, LISS 37°C, IgG

	Donor	Rh-hr								MNSs				P	Lewis		Luth'n		Kell				Duffy		Kidd			Special Type	Test Methods					
		D	C	E	c	e	f	V	C^w	M	N	S	s	P$_1$	Le^a	Le^b	Lu^a	Lu^b	K	k	Kp^a	Js^a	Fy^a	Fy^b	Jk^a	Jk^b	Xg^a		IS	LISS	IgG	RT	4°	
1	A478	o	o	o	+	+	o	o	o	+	+	o	+	+	o	+	o	+	+	+	o	o	o	+	o	+	+		0	0	0			1
2	B102	o	o	o	+	+	o	o	o	+	+	o	+	+	+	o	o	+	o	+	o	o	+	o	+	+	+		2+	0	2+			2
3	C559	o	o	o	+	+	o	o	o	+	o	+	o	+	o	+	o	+	+	+	o	o	+	o	+	o	+		0	0	2+			3
4	D275	o	+	o	+	+	o	o	o	+	+	+	+	o	o	o	o	+	o	+	o	o	o	+	+	+	+		0	0	0			4
5	E164	o	o	+	+	+	o	o	o	o	+	o	+	+	+	o	o	+	o	+	o	o	o	+	+	o	+		2+	0	0			5
6	F065	+	+	o	o	+	o	o	o	+	o	+	+	+	o	+	o	+	o	+	o	o	+	o	+	+	o		0	0	2+			6
7	G163	+	o	o	+	+	o	o	o	+	+	+	o	+	+	o	o	+	o	+	o	o	o	o	+	+	+		2+	0	0			7
8	H168	+	o	+	+	o	o	o	o	o	+	o	+	+	o	+	o	+	o	+	o	o	+	o	o	+	+		0	0	2+			8
9	R331	+	+	+	o	+	o	o	o	+	+	o	+	o	o	+	o	+	o	+	o	o	+	+	+	+	+		0	0	1+			9
10	A624	+	o	o	+	+	o	o	o	+	+	o	+	+	o	o	o	+	o	+	o	o	o	o	+	o	+		0	0	0			10
	AutoCtrl																												0	0	0			
		D	C	E	c	e	f	V	C^w	M	N	S	s	P$_1$	Le^a	Le^b	Lu^a	Lu^b	K	k	Kp^a	Js^a	Fy^a	Fy^b	Jk^a	Jk^b	Xg^a							

Cell #4 does not possess the E antigen. Therefore, even if anti-E was present, it could not possibly react with this cell. Consequently, this is not a good cell for ruling out anti-E.

FIGURE 26.6 Full test panel with intermediate results marked using color coded markers provided by AIDA. In this case, AIDA has interrupted the user to suggest that an error may have been made in ruling out anti-E.

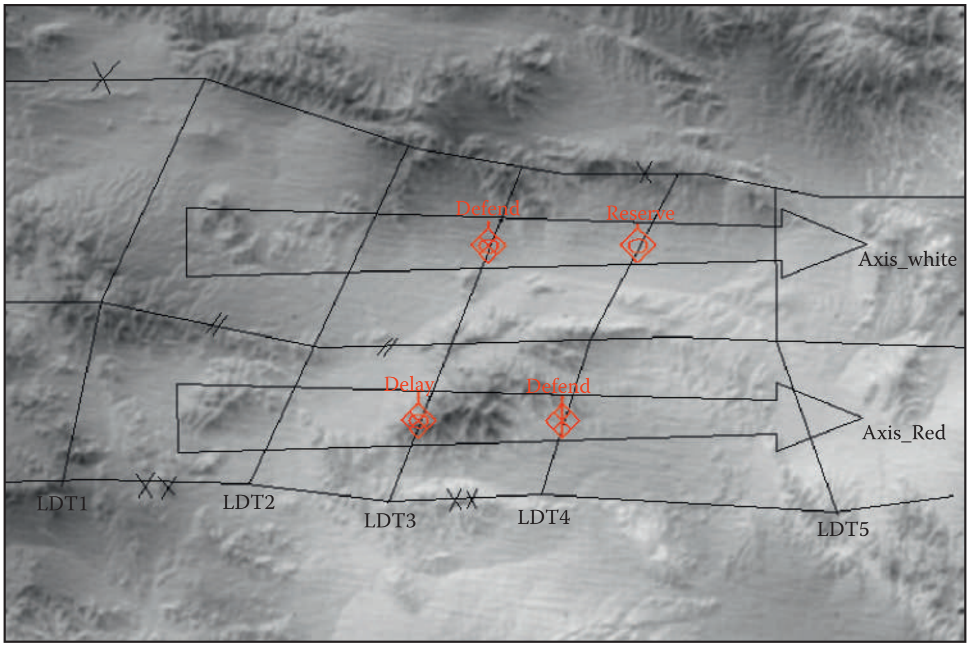
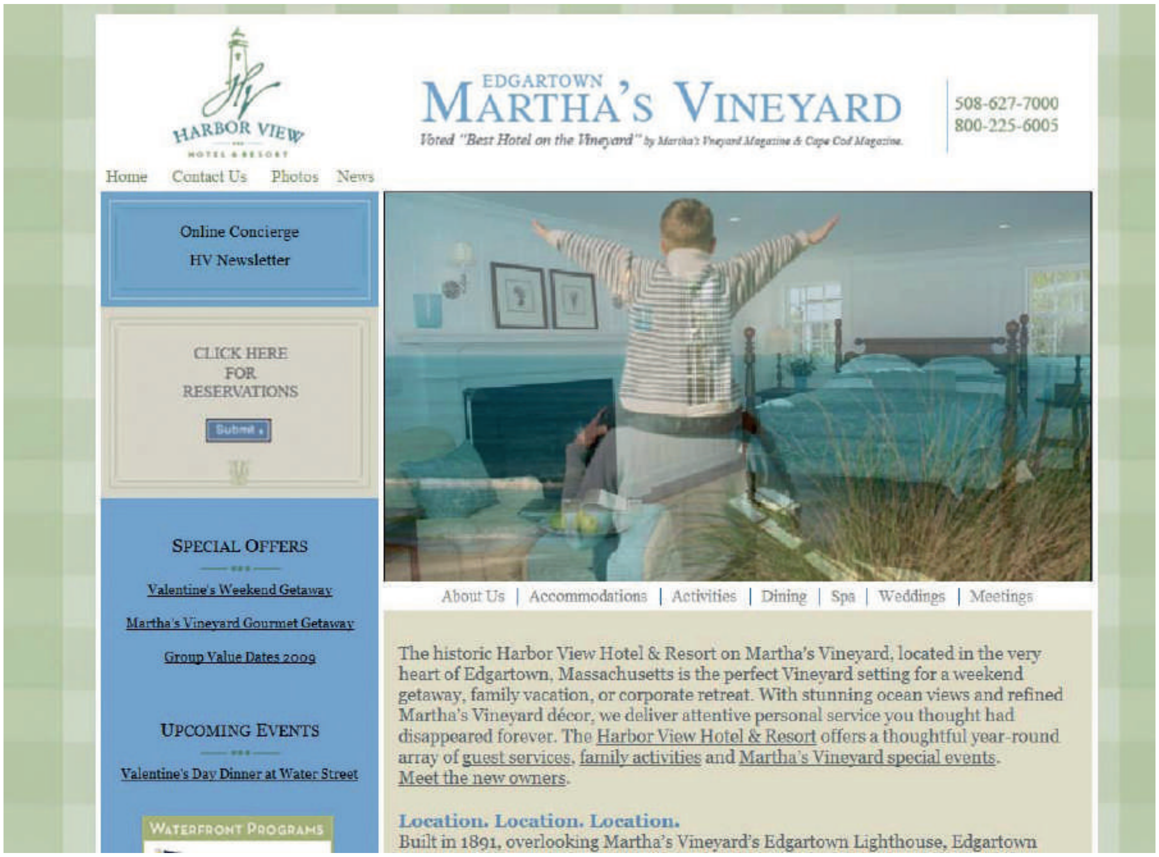


FIGURE 26.7 An example of an enemy course of action (ECOA). The diamond-shaped figures represent the enemy units. Each unit is further labeled by role played by the unit in this scenario: Defend, Reserve, or Delay. The enemy units move along two corridors in the terrain indicated by large arrows and labeled Axis White and Axis Red. Friendly units move in the direction of the arrows, enemy units move in the reverse direction. LDT means Line of Defensible Terrain. LDT 1 through 5 are used as reference lines to mark how far the units have advanced. In this snapshot, all enemy units are currently positioned on LDT 3 and LDT 4.



EDGARTOWN
MARTHA'S VINEYARD
Voted "Best Hotel on the Vineyard" by Martha's Vineyard Magazine & Cape Cod Magazine.

508-627-7000
800-225-6005

Online Concierge
HV Newsletter

CLICK HERE FOR RESERVATIONS
Submit

SPECIAL OFFERS
Valentine's Weekend Getaway
Martha's Vineyard Gourmet Getaway
Group Value Dates 2009

UPCOMING EVENTS
Valentine's Day Dinner at Water Street

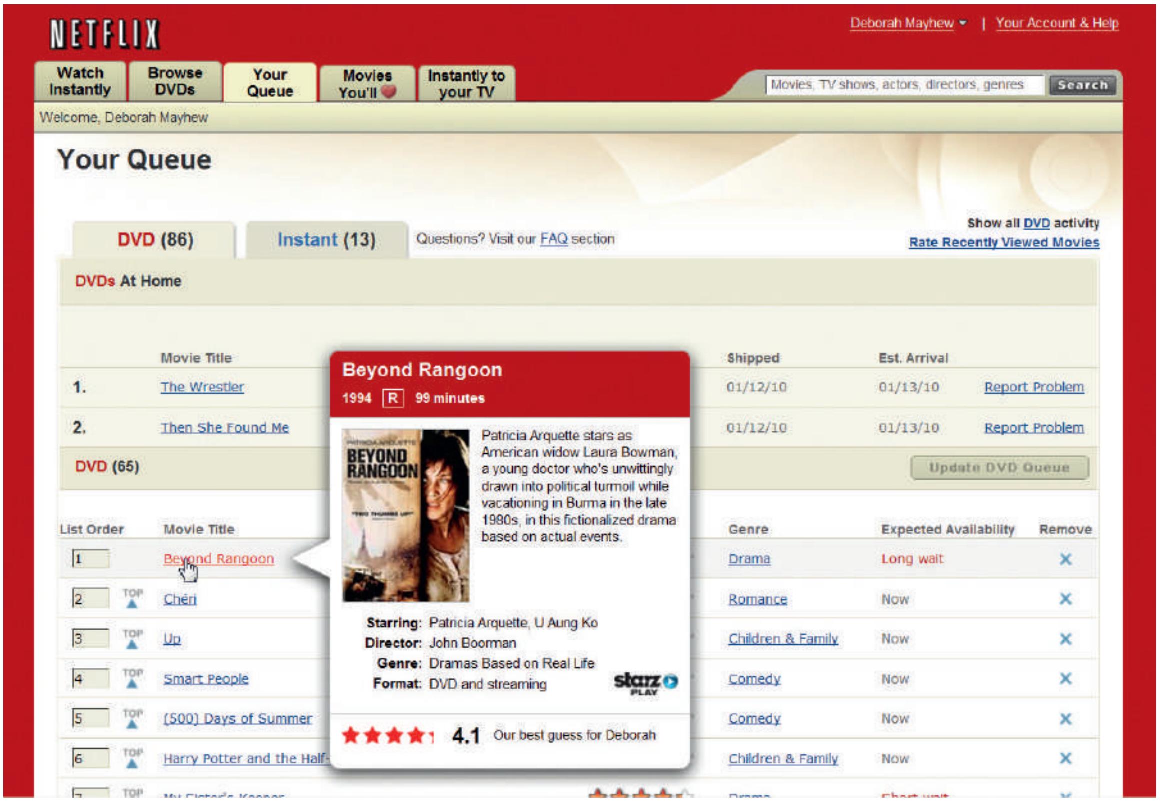
WATERFRONT PROGRAMS

About Us | Accommodations | Activities | Dining | Spa | Weddings | Meetings

The historic Harbor View Hotel & Resort on Martha's Vineyard, located in the very heart of Edgartown, Massachusetts is the perfect Vineyard setting for a weekend getaway, family vacation, or corporate retreat. With stunning ocean views and refined Martha's Vineyard décor, we deliver attentive personal service you thought had disappeared forever. The Harbor View Hotel & Resort offers a thoughtful year-round array of guest services, family activities and Martha's Vineyard special events. Meet the new owners.

Location. Location. Location.
Built in 1891, overlooking Martha's Vineyard's Edgartown Lighthouse, Edgartown

FIGURE 51.1 Usability: invisible functionality. http://www.harbor-view.com/.



Your Queue

DVD (86) Instant (13) Questions? Visit our FAQ section

Show all DVD activity
Rate Recently Viewed Movies

DVDs At Home

	Movie Title	Shipped	Est. Arrival	
1.	The Wrestler	01/12/10	01/13/10	Report Problem
2.	Then She Found Me	01/12/10	01/13/10	Report Problem

DVD (65)

Update DVD Queue

List Order		Movie Title	Genre	Expected Availability	Remove
1		Beyond Rangoon	Drama	Long wait	✕
2	TOP	Chéri	Romance	Now	✕
3	TOP	Up	Children & Family	Now	✕
4	TOP	Smart People	Comedy	Now	✕
5	TOP	(500) Days of Summer	Comedy	Now	✕
6	TOP	Harry Potter and the Half-	Children & Family	Now	✕
7	TOP	My Sister's Keeper	Drama	Short wait	✕

Beyond Rangoon
1994 R 99 minutes

Patricia Arquette stars as American widow Laura Bowman, a young doctor who's unwittingly drawn into political turmoil while vacationing in Burma in the late 1980s, in this fictionalized drama based on actual events.

Starring: Patricia Arquette, U Aung Ko
Director: John Boorman
Genre: Dramas Based on Real Life
Format: DVD and streaming

★★★★ 4.1 Our best guess for Deborah

FIGURE 51.5 Usability: efficiency. http://www.netflix.com/Queue?lnkce=sntQu&lnkctr=mhbque.

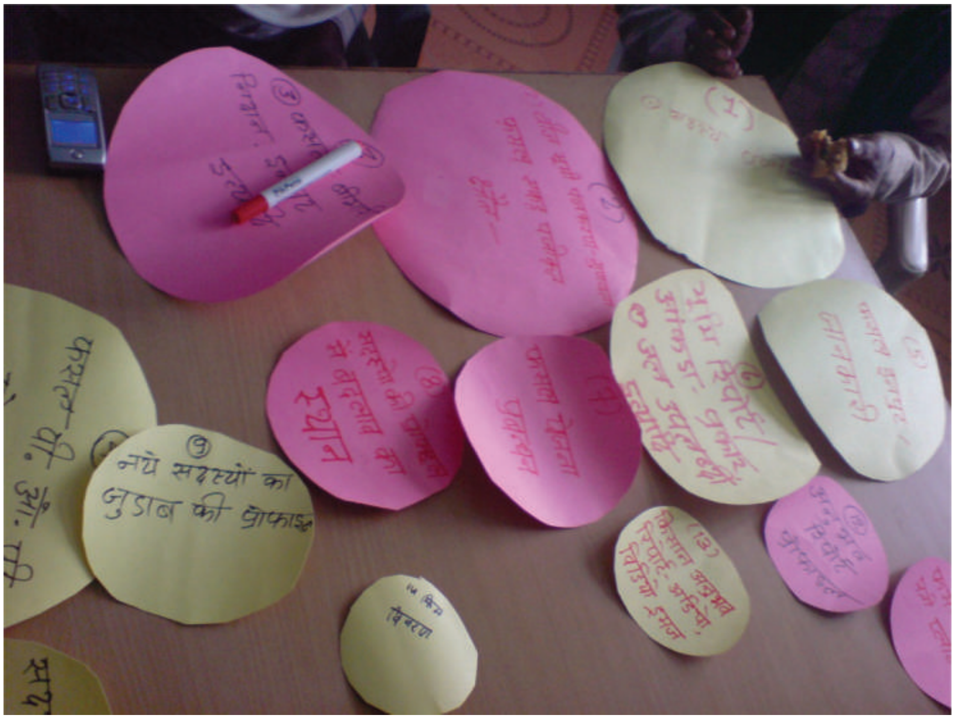


FIGURE 62.1 Example of a chapatti diagram.

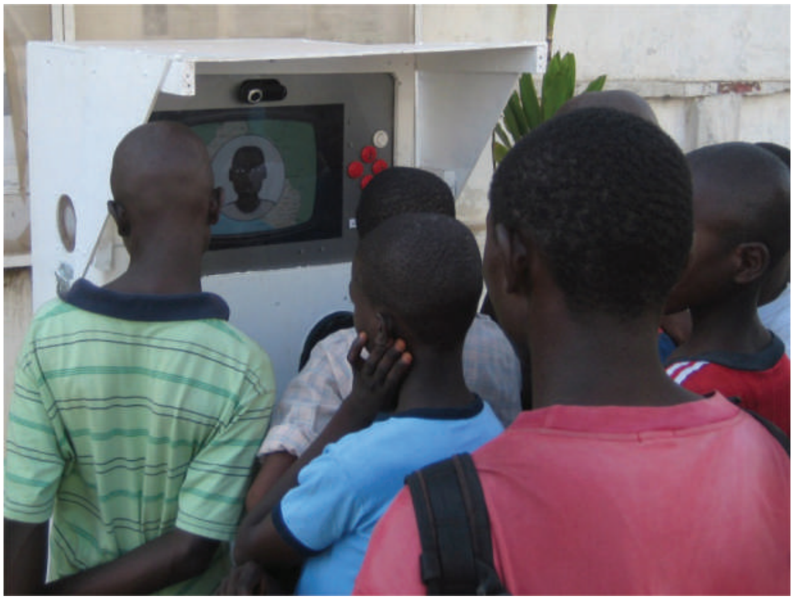
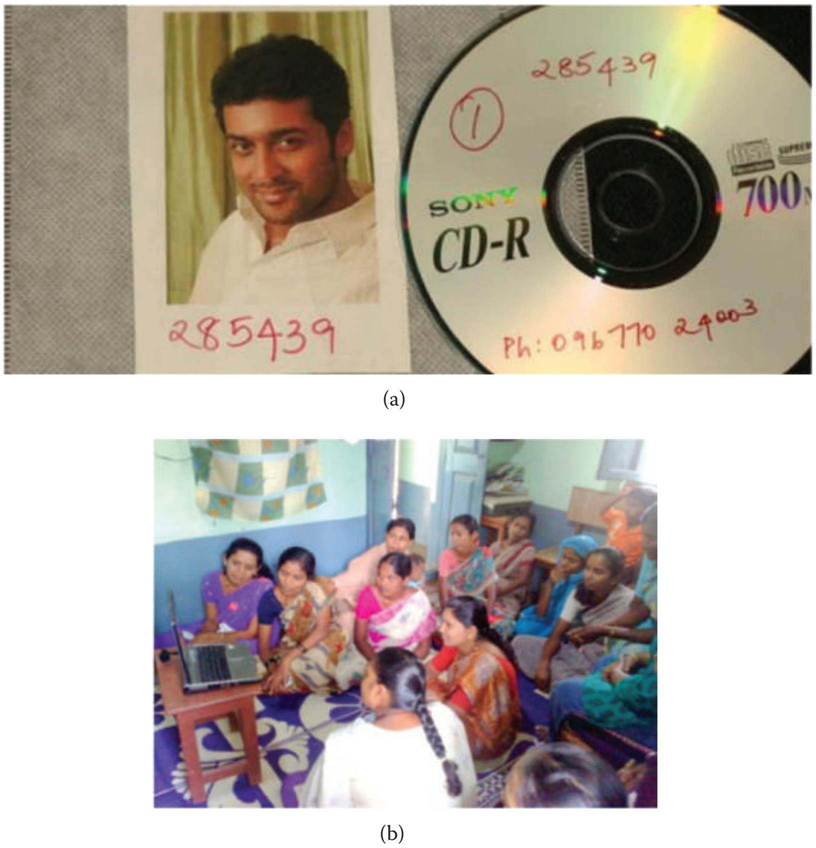


FIGURE 62.2 People using the Mobile Story Exchange System (MOSES) kiosk.



(a)

(b)

FIGURE 62.3 (a) Sample video compact disc and sleeve. (b) A screening session in Jakkur.

Ergonomics and Human Factors

The Human–Computer Interaction Handbook

Fundamentals, Evolving Technologies, and Emerging Applications

Third Edition

"… details the progress of this extraordinary discipline … The carefully written chapters and extensive references will be useful … This handbook's prominent authors thoughtfully survey the key topics, enabling students, researchers, and professionals to appreciate HCI's impact."

—from the foreword by **Ben Shneiderman**, University of Maryland

"If you care about Interaction Design, you should own this book. Exhaustive coverage by world authorities. … one book that covers the gamut. Highly recommended, highly practical."

—**Don Norman**, Northwestern University

"Comprehensive and thorough coverage of all the important issues related to user interfaces and usability. A useful reference work for anybody in the field…"

—**Jakob Nielsen**

The third edition of a groundbreaking reference, **The Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications** raises the bar for handbooks in this field. The book captures the current and emerging sub-disciplines within HCI related to research, development, and practice that continue to advance at an astonishing rate. It features cutting-edge advances to the scientific knowledge base as well as visionary perspectives and developments that fundamentally transform the way in which researchers and practitioners view the discipline.

New and Expanded Topics in the Third Edition

- HCI and global sustainability
- HCI in health care
- Social networks and social media
- Enterprise social computing
- Role of HCI in e-Government
- Role of creativity and cognition in HCI
- Naturalistic approach to evaluation, persuasion, and globalization

The chapter authors include experts from academia, industry, and government agencies from across the globe — all among the very best and most respected in their fields. The more than 80 tables, 400 figures, nearly 7,000 references, and four-page color insert combine to provide the single most comprehensive depiction of this field. Broad in scope, the book pays equal attention to the human side, the computer side, and the interaction of the two. This balanced, application-focused design coverage makes the book not only an excellent research guide but also an authoritative handbook for the practice of HCI and for education and training in HCI.

K11447

CRC Press
Taylor & Francis Group
an informa business
www.crcpress.com

6000 Broken Sound Parkway, NW
Suite 300, Boca Raton, FL 33487
711 Third Avenue
New York, NY 10017
2 Park Square, Milton Park
Abingdon, Oxon OX14 4RN, UK

ISBN: 978-1-4398-2943-1
90000
9 781439 829431